



- DOSSIER -

Contrat du service : Etudiant

| | | | |
|-------------------------|-------------------------------|-------------------|-----|
| Auteur : | AMUE | | |
| Date de création : | 12/04/2012 | Version : | 1.8 |
| Dernière modification : | 07/12/2015 | Nombre de pages : | 49 |
| Destinataires : | Sites utilisateurs | | |
| Pour information : | | | |
| Mots Clés : | CONTRAT WEB SERVICES ETUDIANT | | |
| Accessibilité : | Libre | | |

SUJET : Contrats de service

LISTE DES MODIFICATIONS

| Modification | Version | Emetteur | Validé par | Date |
|---|---------|--------------|-------------|------------|
| Création du document | 0.1 | CapGemini | G.ROSTAIN | 24/04/07 |
| Mise à jour des données sortie | 0.2 | CapGemini | M.ALLEGRE | 14/06/07 |
| Mise en forme après retours AMUE | 0.3 | CapGemini | A. Combe | 07/08/07 |
| Version finale | 1.0 | CapGemini | A. Combe | 25/09/2007 |
| DM49816 & 51560 : Ajout connecteur recupererNumeroSecuriteSociale | 1.1 | Sopra Group | J.Roland | 03/06/2009 |
| WSA 04 : Ajout du connecteur recupererCalendriersExamenEtudiant | 1.2 | Sopra Group | J. Roland | 29/10/2009 |
| DM55236 : Ajout du connecteur recupererListeEtudiants | 1.3 | Sopra Group | Y. Renard | 22/01/2010 |
| DM60505 : Ajout de nouveaux connecteurs | 1.4 | Sopra Group | Y. Renard | 05/07/2010 |
| DM58909 : Mise à jour des wsdl et xsd | 1.4 | Sopra Group | Y. Renard | 17/08/2010 |
| DM62399 : Ajout d'un nouveau connecteur | 1.5 | Sopra Group | Y. Renard | 23/03/2011 |
| DM70761 : Suppression de la version 1 et ajout de la version 3 du connecteur de récupération des calendriers d'examen valides d'un etudiant | 1.6 | Sopra Group | | 02/04/2012 |
| DM83101 : Ajout d'une précision concernant les formats de type date autorisés | 1.7 | Sopra Group | J. Affinito | 17/04/2013 |
| Suppression des connecteurs recupererAdressesEtudiant et recupererCalendriersExamenEtudiant_v2 | 1.8 | Sopra Stéria | | 25/11/2015 |

Modifications depuis la dernière Version

| Paragraphe | Page | Nature de la Modification |
|------------|------|--|
| | | Suppression description du connecteur recupererAdressesEtudiant |
| | | Suppression description du connecteur recupererCalendriersExamenEtudiant_v2 |
| 2.2.7 | 30 | Description du connecteur recupererCalendriersExamenEtudiant_v3 : Ajout du nom de la classe et de la méthode |
| | 2 | Maj sommaire |

TABLE DES MATIERES

| | |
|--|----------|
| 1. OBJECTIFS GENERAUX | 5 |
| 2. SERVICE ETUDIANT | 6 |
| 2.1. PRESENTATION | 6 |
| 2.1.1. OBJECTIFS | 6 |
| 2.1.2. VERSION DU SERVICE | 6 |
| 2.1.3. CAS D'UTILISATION | 6 |
| 2.1.4. TYPE DE SERVICE | 6 |
| 2.1.5. REGLES DE GESTION GENERALE AU SERVICE | 6 |
| 2.1.6. ERREURS / EXCEPTIONS GLOBALES | 6 |
| 2.2. LES OPERATIONS | 7 |
| 2.2.1. CONTROLE DES TYPES D'HEBERGEMENT | 7 |
| 2.2.1.1. <i>Description</i> | 7 |
| 2.2.1.2. <i>Version de l'opération</i> | 7 |
| 2.2.1.3. <i>Paramètres en entrée</i> | 7 |
| 2.2.1.4. <i>Données en sortie</i> | 8 |
| 2.2.1.5. <i>Règles de gestion</i> | 8 |
| 2.2.1.6. <i>Erreurs / Exceptions</i> | 9 |
| 2.2.2. RECUPERATION DES IDENTIFIANTS D'UN ETUDIANT | 10 |
| 2.2.2.1. <i>Description</i> | 10 |
| 2.2.2.2. <i>Version de l'opération</i> | 10 |
| 2.2.2.3. <i>Paramètres en entrée</i> | 10 |
| 2.2.2.4. <i>Données en sortie</i> | 11 |
| 2.2.2.5. <i>Règles de gestion</i> | 11 |
| 2.2.2.6. <i>Erreurs / Exceptions</i> | 12 |
| 2.2.3. LECTURE DES COORDONNEES PERSONNELLES DE L'ETUDIANT V2 | 13 |
| 2.2.3.1. <i>Description</i> | 13 |
| 2.2.3.2. <i>Version de l'opération</i> | 13 |
| 2.2.3.3. <i>Paramètres en entrée</i> | 13 |
| 2.2.3.4. <i>Données en sortie</i> | 14 |
| 2.2.3.5. <i>Règles de gestion</i> | 16 |
| 2.2.3.6. <i>Erreurs / Exceptions</i> | 16 |
| 2.2.4. MODIFICATION DES COORDONNEES PERSONNELLES DE L'ETUDIANT | 17 |
| 2.2.4.1. <i>Description</i> | 17 |
| 2.2.4.2. <i>Version de l'opération</i> | 17 |
| 2.2.4.3. <i>Paramètres en entrée</i> | 17 |
| 2.2.4.4. <i>Données en sortie</i> | 18 |
| 2.2.4.5. <i>Règles de gestion</i> | 18 |
| 2.2.4.6. <i>Erreurs / Exceptions</i> | 20 |
| 2.2.5. RECUPERATION DES INFORMATIONS ADMINISTRATIVES GENERALES DE L'ETUDIANT | 22 |
| 2.2.5.1. <i>Description</i> | 22 |
| 2.2.5.2. <i>Version de l'opération</i> | 22 |
| 2.2.5.3. <i>Paramètres en entrée</i> | 22 |
| 2.2.5.4. <i>Données en sortie</i> | 23 |

| | |
|--|-----------|
| 2.2.5.5. Règles de gestion | 25 |
| 2.2.5.6. Erreurs / Exceptions | 26 |
| 2.2.6. RECUPERATION DU NUMERO DE SECURITE SOCIALE | 26 |
| 2.2.6.1. Description | 26 |
| 2.2.6.2. Version de l'opération | 27 |
| 2.2.6.3. Paramètres en entrée | 27 |
| 2.2.6.4. Données en sortie | 27 |
| 2.2.6.5. Règles de gestion | 27 |
| 2.2.6.6. Erreurs / Exceptions | 27 |
| 2.2.7. RECUPERATION DES CALENDRIERS D'EXAMEN VALIDES D'UN ETUDIANT (VERSION 3) | 28 |
| 2.2.7.1. Description | 28 |
| 2.2.7.2. Version de l'opération | 28 |
| 2.2.7.3. Paramètres en entrée | 28 |
| 2.2.7.4. Données en sortie | 28 |
| 2.2.7.5. Règles de gestion | 30 |
| 2.2.7.6. Erreurs / Exceptions | 30 |
| 2.2.8. RECUPERATION DE LA LISTE DES ETUDIANTS | 32 |
| 2.2.8.1. Description | 32 |
| 2.2.8.2. Version de l'opération | 32 |
| 2.2.8.3. Paramètres en entrée | 32 |
| 2.2.8.4. Données en sortie | 33 |
| 2.2.8.5. Règles de gestion | 34 |
| 2.2.8.6. Erreurs / Exceptions | 43 |
| 2.3. CONTRAT WSDL | 47 |
| 2.3.1. CONTRAT D'IMPLEMENTATION | 47 |
| 2.3.2. CONTRAT ABSTRAIT | 47 |
| 2.3.3. SCHEMAS XML DES DONNEES D'ENTREE/SORTIE | 47 |
| 3. ELEMENTS COMMUNS | 47 |

1.OBJECTIFS GENERAUX

Dans un contexte de forte augmentation du besoin d'interopérabilité, de communication au sein du système d'information et d'ouverture applicative, il est nécessaire d'ouvrir l'application Apogée. C'est pour ces raisons, et au travers de ce contexte que le choix technologique des Web Services s'est imposé comme étant la solution idéale.

Le service de la partie étudiant du dossier étudiant est très important car il permet de traiter les données individuelles de l'étudiant présentes dans Apogée.

L'objectif principal de ce type de composant est de fournir des listes et/ou des contrôles relatifs au dossier étudiant.

2.SERVICE ETUDIANT

2.1. PRESENTATION

2.1.1. OBJECTIFS

Les objectifs du service autour de la partie « étudiant » du dossier étudiant d'Apogée sont :

- Ouvrir Apogée au système d'information
- Fournir l'ensemble des données pertinentes sur cette partie pour le système d'information
- Fournir une solution technique interopérable
- Casser l'adhérence des développements locaux vis à vis d'Apogée
- Faciliter aux travers de ces connecteurs l'accès à une information garantie

2.1.2. VERSION DU SERVICE

| Version | Principales évolutions | Date |
|---------|---|------------|
| 0.1 | | 11/05/07 |
| 1.0 | Ajout du connecteur recupererCalendriersExamenEtudiant | 28/10/2009 |
| 1.1 | Ajout du connecteur recupererListeEtudiants | 22/01/2010 |
| 1.2 | Suppression de la version 1 et ajout de la version 3 du connecteur de récupération des calendriers d'examen valides d'un etudiant | 31/05/2012 |

2.1.3. CAS D'UTILISATION

Voici une liste non exhaustive des cas d'utilisation possible de ce service :

- Utilisation par les applications du système d'information pour accéder au référentiel individu d'Apogée
- Réutilisation par des services numériques d'Apogée

Attention, ces connecteurs sont susceptibles de retourner des données sensibles. Il est de la responsabilité du programme appelant de gérer correctement l'utilisation et la diffusion de ces données.

2.1.4. TYPE DE SERVICE

Il s'agit d'un service de type Web Service synchrone (question / réponse). Ce type de service peut être utilisé pour des besoins synchrones mais peut également être utilisé dans des cas d'utilisation asynchrones.

2.1.5. REGLES DE GESTION GENERALE AU SERVICE

Sans Objets.

2.1.6. ERREURS / EXCEPTIONS GLOBALES

Toutes les méthodes Java des services peuvent générer des « runtime » exception de type WebBaseException.

En cas de problème lors de l'appel des Web Services, il faut pouvoir communiquer l'information à l'appelant et donc incorporer au sein du flux XML de réponse du Web Service l'exception levée. Cela est réalisé grâce à l'utilisation du mécanisme d'erreur (élément *fault*) proposé par les technologies Web Service.

Lors de l'appel à distance des méthodes de la couche services, les exceptions levées sont transformées en élément de type *fault* et transmises à l'appelant dans le flux XML de réponse.

2.2. LES OPERATIONS

2.2.1. CONTROLE DES TYPES D'HEBERGEMENT

Cette opération est liée à la méthode `recupererTypeHebergement` de la classe `EtudiantMetierServiceInterface` du stub client d'appel fourni par l'AMUE. La documentation de cette classe Java est disponible sous forme de JavaDoc.

2.2.1.1. Description

Ce connecteur a pour objectif de retourner l'ensemble des types d'hébergements du référentiel Apogée.

2.2.1.2. Version de l'opération

| Version | Principales évolutions | Date |
|---------|------------------------|----------|
| 0.1 | | 11/05/07 |

2.2.1.3. Paramètres en entrée

Ce chapitre décrit de façon détaillée les paramètres en entrée sous la forme d'un tableau :

| Nom de la donnée | Description | Nom technique | Type | Obligatoire |
|---|--|-----------------------------|------------|-------------|
| Code du type hébergement | Code du type d'hébergement que le connecteur doit retourner | <code>_code</code> | String (8) | N |
| Récupération avec témoin en service activé (O) ou non (N) ou tous (T) | Paramètre servant de filtre sur le témoin en service du type d'hébergement | <code>_temoinService</code> | String (1) | N |
| Récupération avec témoin accès à distance activé (O) ou non (N) ou tous (T) | Paramètre servant de filtre sur le témoin accès à distance du type d'hébergement | <code>_temoinAD</code> | String (1) | N |

2.2.1.4. Données en sortie

Ce chapitre décrit de façon détaillée les données en sortie sous la forme d'un tableau :

| Nom de la donnée | Description | Nom technique | Type | Obligatoire |
|-------------------------------|------------------------------------|-------------------------|----------------------|-------------|
| Liste des types d'hébergement | Liste des types d'hébergement | - | TypeHebergementDTO[] | N |
| Code | Code du type d'hébergement | codTypeHebergement | String | O |
| Libellé long | Libellé long du type d'hébergement | libTypeHebergement | String | N |
| Libellé web | Libellé web du type d'hébergement | libWebTypeHebergement | String | N |
| Témoin En Service | Témoin « En Service » | temEnSveTypeHebergement | String | O |
| Témoin Accès à Distance | Témoin « Accès à Distance » | temMinTypeHebergement | String | O |

2.2.1.5. Règles de gestion

- Paramètre code du type d'hébergement

Le Web Service retourne les données de la table des types d'hébergement du référentiel d'Apogée correspondant au code du type d'hébergement passé en paramètre.

Si ce paramètre est vide, le Web Service renvoie l'ensemble des types d'hébergement du référentiel Apogée (en tenant compte des autres filtres potentiels).

- Paramètre Témoin En Service

Le Web Service retourne les données de la table des types d'hébergement du référentiel d'Apogée correspondant au témoin en service passé en paramètre.

Ce témoin peut prendre plusieurs valeurs :

- « O » : le Web Service renvoie uniquement les types d'hébergement en service
- « N » : le Web Service renvoie uniquement les types d'hébergement hors service
- « T » : le Web Service renvoie tous les types d'hébergement sans tenir compte du témoin

Si ce paramètre est vide, le Web Service renvoie l'ensemble des types d'hébergement en service. La valeur par défaut est donc « O ».

Si ce paramètre ne contient pas une de ces valeurs ou n'est pas vide alors le Web Service remonte une erreur précise exprimant le fait que le paramètre est incorrect.

- Paramètre Témoin Accès à Distance

Le Web Service retourne les données de la table des types d'hébergement du référentiel d'Apogée correspondant au témoin accès à distance passé en paramètre.

Ce témoin peut prendre plusieurs valeurs :

- « O » : le Web Service renvoie uniquement les types d’hébergement dont le témoin AD est coché
- « N » : le Web Service renvoie uniquement les types d’hébergement dont le témoin AD est décoché
- « T » : le Web Service renvoie tous les types d’hébergement sans tenir compte du témoin

Si ce paramètre est vide, le Web Service renvoie l’ensemble des types d’hébergement sans tenir compte du témoin. La valeur par défaut est donc « Tous ».

Si ce paramètre ne contient pas une de ces valeurs ou n’est pas vide alors le Web Service remonte une erreur précise exprimant le fait que le paramètre est incorrect.

- Récupération des types d’hébergement

Si aucun type d’hébergement correspondant aux critères n’est trouvé dans Apogée, le Web Service retourne une erreur précise exprimant qu’il n’a pas trouvé un seul type d’hébergement dans Apogée correspondant aux critères.

- Les libellés

Le libellé sera le libellé long de la table d’Apogée ainsi que le libellé web correspondant (si il existe).

- Témoin En Service

Concernant le champ « Témoin En Service » (O ou N), il n’y aura pas de transformation : le Web Service retourne O ou N.

- Témoin Accès à Distance

Concernant le champ « Témoin Accès à Distance » (O ou N), il n’y aura pas de transformation : le Web Service retourne O ou N.

2.2.1.6. Erreurs / Exceptions

En complément aux erreurs/exceptions ci-dessous, il faut ajouter les erreurs/exceptions du chapitre 2.1.6 ainsi que celles qui sont communes à l’ensemble des Web Services (voir § 3).

| Erreur | Type | Description |
|---|---|---|
| Aucun type d’hébergement n’a été trouvé | technical.data.nullretrieve.typeHebergement | La recherche n’a pas trouvé un seul type d’hébergement correspondant aux critères |
| Paramètre témoin en service incorrect | technical.parameter.noncoherentinput.temoindenService | Le paramètre ne contient pas une valeur autorisée |
| Paramètre témoin Accès à Distance incorrect | technical.parameter.noncoherentinput.temoindenAD | Le paramètre AD ne contient pas une valeur autorisée |

2.2.2. RECUPERATION DES IDENTIFIANTS D'UN ETUDIANT

Cette opération est liée à la méthode `recupererIdentifiantsEtudiant` de la classe `EtudiantMetierServiceInterface` du stub client d'appel fournit par l'AMUE. La documentation de cette classe Java est disponible sous forme de JavaDoc.

2.2.2.1. Description

Cette opération doit permettre de récupérer l'ensemble des identifiants d'un étudiant dans la base de données Apogée.

2.2.2.2. Version de l'opération

| Version | Principales évolutions | Date |
|---------|------------------------|----------|
| 0.1 | | 11/05/07 |

2.2.2.3. Paramètres en entrée

| Nom de la donnée | Description | Nom technique | Type | Obligatoire |
|---------------------------------|--|---------------------------|-------------|-------------|
| Code étudiant | Numéro de l'étudiant | <code>_codEtu</code> | String (8) | N |
| Code individu | Code interne Apogée de l'étudiant | <code>_codInd</code> | String (8) | N |
| Numéro INE étudiant | Numéro identifiant national de l'étudiant | <code>_numINE</code> | String (10) | N |
| Clé INE | Clef INE | <code>_cleINE</code> | String (1) | N |
| Numéro boursier étudiant | Le numéro de boursier unique de l'étudiant | <code>_numBoursier</code> | String (13) | N |
| Code OPI | Le code Opération Préalable à l'Inscription de l'étudiant | <code>_codOPI</code> | String (8) | N |
| Nom de l'étudiant | Nom patronymique de l'étudiant | <code>_nom</code> | String (30) | N |
| Prénom de l'étudiant | Prénom de l'étudiant | <code>_prenom</code> | String (20) | N |
| Date de naissance de l'étudiant | Date de naissance de l'étudiant (formats autorisés : DDMMYY, DDMMYYYY, DD/MM/YY, DD/MM/YYYY) | <code>_dateNaiss</code> | Date | N |

| | | | | |
|-----------------------|--|------------------|------------|---|
| Récupération annuaire | Ce paramètre (Oui ou Non) permet de préciser si on souhaite utiliser le package Apogée PKB_ANNU pour aller récupérer le login et l'adresse email dans l'annuaire (ceci sous réserve du paramétrage dans le référentiel). | _temoinRecupAnnu | String (1) | N |
|-----------------------|--|------------------|------------|---|

2.2.2.4. Données en sortie

Ce chapitre décrit de façon détaillée les données en sortie sous la forme d'un tableau :

| Nom de la donnée | Description | Nom technique | Type | Obligatoire |
|---------------------------|---|------------------|-------------------------|-------------|
| Identifiant de l'étudiant | Identifiant de l'étudiant | - | IdentifiantsEtudiantDTO | O |
| Code étudiant | Code étudiant au sein de l'établissement | codEtu | Integer | O |
| Code individu | Code étudiant au sein d'Apogée | codInd | Integer | O |
| Numéro INE étudiant | Numéro national de l'étudiant (avec la clé) | numeroINE/cleINE | String | O |
| Numéro boursier étudiant | Numéro boursier étudiant | numBoursier | String | N |
| Login annuaire | Login annuaire | loginAnnuaire | String | N |
| Email annuaire | Email annuaire | emailAnnuaire | String | N |

2.2.2.5. Règles de gestion

- Les paramètres

Le Web Service prend en paramètre soit :

- Le code étudiant correspondant au numéro étudiant d'Apogée
- Le code individu de l'étudiant correspondant au code interne d'Apogée
- Le numéro INE et sa clé
- Le numéro de boursier de l'étudiant
- Le code OPI de l'étudiant
- L'identité de l'étudiant correspondant au nom patronymique, au prénom et la date de naissance de l'étudiant.

Si aucun de ces paramètres n'est renseigné alors le Web Service remonte une erreur précise exprimant le fait qu'il faut au moins un paramètre.

Si plus d'un de ces 5 paramètres a été envoyé alors le Web Service remonte une erreur exprimant le fait qu'il y a trop de paramètres.

Dans le cas d'une recherche par le groupe de paramètres de l'identité de l'étudiant, il faut que les 3 paramètres soient présents. Si ce n'est pas le cas, le Web Service remonte une erreur précise exprimant le fait que le nom, le prénom et la date de naissance sont indissociables.

Le paramètre « récupération annuaire » permet de préciser si on souhaite récupérer des données dans l'annuaire à partir de l'interface fournit par l'AMUE (package PKB_ANNU). Deux valeurs sont possibles :

- « O » pour signifier « Oui ». Sous réserve du paramétrage dans le référentiel Apogée de cette fonctionnalité, le Web Service utilise la procédure correspondante pour récupérer le login et l'adresse email.
- « N » pour signifier « Non ». Dans ce cas là, quelque soit le paramétrage dans le référentiel Apogée, le Web Service n'utilise pas cette fonctionnalité et ne récupérera donc pas de login ni d'adresse email dans l'annuaire.

Si ce paramètre est vide, la valeur par défaut sera celle du référentiel Apogée : si l'accès à l'annuaire est paramétré alors la valeur par défaut sera « O », sinon la valeur par défaut sera « N »

Si le paramètre est à « O » et que le paramétrage dans le référentiel n'est pas fait alors le Web Service remonte une erreur précise exprimant le fait qu'il y a une incohérence entre la donnée en entrée et le paramétrage dans le référentiel.

- La recherche

Si les paramètres sont cohérents, le Web Service recherche l'étudiant correspondant au paramètre envoyé. Si aucun étudiant ne correspond, le Web Service retourne une erreur précisant qu'aucun étudiant n'a été trouvé. Si plusieurs étudiants correspondent, le Web Service retourne une erreur précisant le fait qu'il n'existe pas un étudiant unique correspondant à la recherche.

Le Web Service retourne l'ensemble des données attendues en retour. Concernant la donnée « Code INE », le Web Service retourne une concaténation de l'INE et de sa clé.

En fonction du paramètre « récupération annuaire » et du paramétrage dans le référentiel Apogée, le connecteur appelle la procédure du package PKB_ANNU pour récupérer le login et l'adresse email de l'étudiant dans l'annuaire.

Il est à noter que le connecteur ne retourne aucune donnée nominative de l'étudiant. Il ne servira donc jamais à de l'identification nominative à partir d'un code.

- Cas particulier : le code OPI

La recherche d'un étudiant par son code OPI ne se fera que pour les étudiants véritablement inscrit dans Apogée. Ainsi, la recherche se fera par le champ COD_OPI_INT_EPO de la table IND_OPI sous réserve qu'il y ai bien un étudiant (INDIVIDU) qui dispose de cette OPI (jointure avec le champ Apogée COD_IND_OPI) des deux tables.

2.2.2.6. Erreurs / Exceptions

En complément aux erreurs/exceptions ci-dessous, il faut ajouter les erreurs/exceptions du chapitre 2.1.6 ainsi que celles qui sont communes à l'ensemble des Web Services (voir § 3).

| Erreur | Type | Description |
|-------------------------------|---|---|
| Aucun étudiant n'a été trouvé | <code>technical.data.nullretrieve.etudiant</code> | La recherche n'a pas trouvé l'étudiant correspondant |
| Etudiant non unique | <code>technical.data.toomuchretrieve.etudiant</code> | Il a été trouvé plus d'un étudiant correspondant aux paramètres |
| Il faut au moins un paramètre | <code>technical.parameter.noncoherentinput.limiteinf</code> | Le connecteur a été appelé avec aucun paramètre |
| Trop de paramètres | <code>technical.parameter.noncoherentinput.limitesup</code> | Trop de paramètres sont envoyés au connecteur. Il faut en envoyer un et un seul (sauf le cas de l'identité qui en regroupe trois) |

2.2.3. LECTURE DES COORDONNEES PERSONNELLES DE L'ETUDIANT V2

Cette opération est liée à la méthode `recupererAdressesEtudiant_v2` de la classe `EtudiantMetierServiceInterface` du stub client d'appel fourni par l'AMUE. La documentation de cette classe Java est disponible sous forme de JavaDoc.

2.2.3.1. Description

Ce connecteur a pour objectif de retourner l'adresse fixe et l'adresse annuelle (pour l'année demandée) de l'étudiant.

2.2.3.2. Version de l'opération

| Version | Principales évolutions | Date | Date de fin de maintenance |
|---------|------------------------|------------|----------------------------|
| 0.1 | Création du connecteur | 23/03/2011 | |

2.2.3.3. Paramètres en entrée

| Nom de la donnée | Description | Nom technique | Type | Obligatoire |
|-----------------------|---|-----------------------------|------------|-------------|
| Code étudiant | Numéro de l'étudiant. | <code>_codEtu</code> | String (8) | O |
| Année | Année pour l'adresse annuelle de l'étudiant | <code>_annee</code> | String (8) | N |
| Récupération annuaire | Ce paramètre (Oui ou Non) permet de préciser si on souhaite utiliser le package Apogée PKB_ANNU pour aller récupérer le | <code>_recupAnnuaire</code> | String (1) | N |

| | | | | |
|--|--|--|--|--|
| | login et l'adresse email dans l'annuaire (ceci sous réserve du paramétrage dans le référentiel). | | | |
|--|--|--|--|--|

2.2.3.4. Données en sortie

Ce chapitre décrit de façon détaillée les données en sortie sous la forme d'un tableau :

| Nom de la donnée | Description | Nom technique | Type | Obligatoire |
|---------------------------|--|---------------|-----------------|-------------|
| Coordonnées de l'étudiant | Coordonnées de l'étudiant | - | CoordonneesDT02 | N |
| Adresse fixe | Adresse fixe | adresseFixe | AdresseDT02 | O |
| Libelle 1 | Première partie du libellé d'adresse fixe | libAd1 | String | O |
| Libelle 2 | Deuxième partie du libellé d'adresse fixe | libAd2 | String | N |
| Libelle 3 | Troisième partie du libellé d'adresse fixe | libAd3 | String | N |
| Commune | Commune de l'adresse fixe | commune | CommuneDT02 | N |
| Code postal | Code postal de l'adresse fixe | nomCommune | String | O |
| Code commune | Code commune de l'adresse fixe | codePostal | String | O |
| Libellé commune | Libellé commune de l'adresse fixe | nomCommune | String | N |
| Libellé acheminement | Libellé acheminement de l'adresse annuelle | libAch | String | N |
| Pays | Pays de l'adresse fixe | pays | PaysDT0 | O |
| Code pays | Code pays de l'adresse fixe | codPay | String | O |
| Libellé pays | Libellé pays de l'adresse fixe | libPay | String | N |
| Acheminement étranger | Libellé d'adresse annuelle à l'étranger | libAde | String | N |
| Téléphone | Numéro de téléphone pour l'adresse fixe | numTel | String | N |

| | | | | |
|-------------------------------------|--|-----------------|-------------------------|---|
| Adresse annuelle | Adresse annuelle | adresseAnnuelle | AdresseDT02 | N |
| Libelle 1 | Première partie du libellé d'adresse annuelle | libAd1 | String | O |
| Libelle 2 | Deuxième partie du libellé d'adresse annuelle | libAd2 | String | N |
| Libelle 3 | Troisième partie du libellé d'adresse annuelle | libAd3 | String | N |
| Commune | Commune de l'adresse annuelle | commune | CommuneDT02 | N |
| Code postal | Code postal de l'adresse annuelle | nomCommune | String | O |
| Code commune | Code commune de l'adresse annuelle | codePostal | String | O |
| Libellé commune | Libellé commune de l'adresse annuelle | nomCommune | String | N |
| Libellé acheminement | Libellé acheminement de l'adresse annuelle | libAch | String | N |
| Pays | Pays de l'adresse annuelle | pays | PaysDT0 | O |
| Code pays | Code pays de l'adresse annuelle | codPay | String | O |
| Libellé pays | Libellé pays de l'adresse annuelle | libPay | String | N |
| Acheminement étranger | Libellé d'adresse annuelle à l'étranger | libAde | String | N |
| Téléphone | Numéro de téléphone pour l'adresse annuelle | numTel | String | N |
| Téléphone portable | Numéro de téléphone portable | numTelPortable | String | N |
| Adresse e-mail | Adresse e-mail personnelle de l'étudiant | email | String | N |
| Login de l'étudiant dans l'annuaire | Login de l'étudiant dans l'annuaire | loginAnnuaire | String | N |
| Email de l'étudiant dans l'annuaire | Email de l'étudiant dans l'annuaire | emailAnnuaire | String | N |
| Année adresse annuelle | Année adresse annuelle | annee | String | N |
| Type hébergement adresse annuelle | Type hébergement adresse annuelle | typeHebergement | TypeHebergementCourtDT0 | N |

| | | | | |
|------------------------------|---|-----------------------|--------|---|
| Code type hébergement | Code type hébergement adresse annuelle | codTypeHebergement | String | O |
| Libellé type hébergement | Libellé type hébergement adresse annuelle | libTypeHebergement | String | N |
| Libellé web type hébergement | Libellé web type hébergement adresse annuelle | libWebTypeHebergement | String | N |

2.2.3.5. Règles de gestion

- Paramètre numéro étudiant

Il s'agit bien du COD_ETU présent dans Apogée.

Si l'étudiant n'existe pas, alors, le Web Service remonte une erreur précise exprimant le fait que l'étudiant n'existe pas.

- Paramètre année

Si l'étudiant n'a pas une inscription administrative annuelle pour l'année demandée alors, le Web Service remonte une erreur précise exprimant le fait que l'étudiant n'a jamais eu une inscription dans Apogée pour l'année passée en paramètre.

L'année étant facultative, s'il n'y a pas d'année passée en paramètre alors, le Web Service ramène l'adresse fixe et l'adresse annuelle la plus récente. Si il n'y a jamais eu une inscription administrative annuelle alors, le Web Service remonte une erreur précise exprimant le fait que l'étudiant n'a jamais eu d'inscription.

- Paramètre récupération annuaire

Le paramètre « récupération annuaire » permet de préciser si on souhaite récupérer des données dans l'annuaire à partir de l'interface fournit par l'AMUE (package PKB_ANNU). Deux valeurs sont possibles :

- « O » pour signifier « Oui ». Sous réserve du paramétrage dans le référentiel Apogée de cette fonctionnalité, le Web Service utilise la procédure correspondante pour récupérer le login et l'adresse email.
- « N » pour signifier « Non ». Dans ce cas là, quelque soit le paramétrage dans le référentiel Apogée, le Web Service n'utilise pas cette fonctionnalité et ne récupérera donc pas de login ni d'adresse email dans l'annuaire.

Si ce paramètre est vide, la valeur par défaut sera celle du référentiel Apogée : si l'accès à l'annuaire est paramétré alors la valeur par défaut sera « O », sinon la valeur par défaut sera « N »

Si le paramètre est à « O » et que le paramétrage dans le référentiel n'est pas fait alors le Web Service remonte une erreur précise exprimant le fait qu'il y a une incohérence entre la donnée en entrée et le paramétrage dans le référentiel.

2.2.3.6. Erreurs / Exceptions

En complément aux erreurs/exceptions ci-dessous, il faut ajouter les erreurs/exceptions du chapitre 2.1.6 ainsi que celles qui sont communes à l'ensemble des WebServices (voir § 3).

| Erreur | Type | Description |
|---|---|--|
| L'étudiant n'existe pas | technical.security.invaliduser.user | L'étudiant n'a pas été trouvé dans la base de données Apogée |
| Pas d'inscription | technical.data.nullretrieve.findIAA | Il n'existe pas dans Apogée une inscription administrative annuelle pour l'année demandée ou qu'il n'existe pas dans Apogée une seule inscription administrative annuelle (si l'année n'est pas passée en paramètre) |
| Incohérence entre le paramètre récupération annuaire et le paramétrage Apogée | technical.parameter.noncoherentinput.etudiant | L'appelant veut récupérer le login et l'email dans l'annuaire alors que le paramétrage dans Apogée ne le permet pas |

2.2.4. MODIFICATION DES COORDONNEES PERSONNELLES DE L'ETUDIANT

Cette opération est liée à la méthode mettreAJourAdressesEtudiant de la classe EtudiantMetierServiceInterface du stub client d'appel fourni par l'AMUE. La documentation de cette classe Java est disponible sous forme de JavaDoc.

2.2.4.1. Description

Ce connecteur a pour objectif de modifier l'adresse fixe et l'adresse annuelle (pour l'année demandée) de l'étudiant.

2.2.4.2. Version de l'opération

| Version | Principales évolutions | Date |
|---------|------------------------|----------|
| 0.1 | | 11/05/07 |

2.2.4.3. Paramètres en entrée

| Nom de la donnée | Nom technique | Type | Obligatoire |
|---------------------------|---------------|----------------|-------------|
| Code étudiant | _codEtu | String | O |
| Coordonnées de l'étudiant | _coordonnees | CoordonneesDTO | O |
| Adresse fixe | adresseFixe | AdresseMajDTO | O |
| Adresse1 fixe | libAd1 | String | O |
| Adresse2 fixe | libAd2 | String | N |
| Adresse3 fixe | libAd3 | String | N |
| Commune fixe | commune | CommuneMajDTO | N |
| Code postal fixe | codePostal | String | N |
| Code commune fixe | codeInsee | String | N |

| | | | |
|--|-----------------|---------------|---|
| Code pays fixe | codPays | String | O |
| Acheminement étranger fixe | libAde | String | N |
| Téléphone fixe | numTel | String | N |
| Adresse annuelle | adresseAnnuelle | AdresseMajDTO | O |
| Adresse1 annuelle | libAd1 | String | O |
| Adresse2 annuelle | libAd2 | String | N |
| Adresse3 annuelle | libAd3 | String | N |
| Commune annuelle | commune | CommuneMajDTO | N |
| Code postal annuelle | codePostal | String | N |
| Code commune annuelle | codeInsee | String | N |
| Code pays annuelle | codPays | String | O |
| Acheminement étranger annuelle | libAde | String | N |
| Téléphone annuelle | numTel | String | N |
| Téléphone portable | numTelPortable | String | N |
| Adresse e-mail personnelle de l'étudiant | email | String | N |
| Année adresse annuelle | annee | String | O |
| Code type hébergement adresse annuelle | typeHebergement | String | O |

2.2.4.4. *Données en sortie*

Sans objet.

2.2.4.5. *Règles de gestion*

- Adresse fixe et adresse annuelle

Le connecteur a pour objectif de modifier l'adresse fixe et l'adresse annuelle en même temps. Il attend donc en paramètres les deux adresses pour effectuer une modification globale. Si les deux adresses ne sont pas présentes, il devra faire remonter une erreur précise.

Ainsi, pour ne modifier que l'adresse fixe par exemple, il faudra tout de même passer en paramètre une adresse annuelle.

- Paramètre numéro étudiant

Il s'agit bien du COD_ETU présent dans Apogée.

Si l'étudiant n'existe pas, alors, le Web Service remonte une erreur précise exprimant le fait que l'étudiant n'existe pas.

- Paramètre année adresse annuelle

Si l'étudiant n'a pas une inscription administrative annuelle pour l'année demandée alors, le Web Service remonte une erreur précise exprimant le fait que l'étudiant n'a jamais eu d'inscription dans Apogée pour l'année passée en paramètre.

Sinon, si l'étudiant a bien une inscription administrative annuelle pour l'année demandée alors, le connecteur modifie l'unique adresse fixe et l'adresse annuelle de l'année passé en paramètre.

L'année étant facultative, si il n'y a pas d'année passée en paramètre alors, le Web Service modifie l'unique adresse fixe et l'adresse annuelle la plus récente. Si il n'y a jamais eu une inscription administrative annuelle alors, le Web Service remonte une erreur précise exprimant le fait que l'étudiant n'a jamais eu d'inscription dans Apogée.

- Paramètre type d'hébergement

Ce paramètre est obligatoire pour la mise à jour des adresses. S'il n'est pas présent le Web Service remonte une erreur précise exprimant le fait que ce paramètre est obligatoire.

Le Web Service vérifie que le code passé dans ce paramètre existe dans la table du référentiel d'Apogée correspondante. Si ce code n'existe pas ou est lié à une donnée hors service, le Web Service remonte une erreur précise exprimant le fait que le code est invalide.

- Paramètre Adresse1 fixe

Ce paramètre est obligatoire. S'il n'est pas présent ou vide, le Web Service remonte une erreur précise exprimant le fait que ce paramètre est obligatoire.

- Paramètre Adresse1 annuelle

Ce paramètre est obligatoire. S'il n'est pas présent ou vide, le Web Service remonte une erreur précise exprimant le fait que ce paramètre est obligatoire.

- Paramètre Pays fixe

Ce paramètre est obligatoire. S'il n'est pas présent ou vide, le Web Service remonte une erreur précise exprimant le fait que ce paramètre est obligatoire.

Le Web Service vérifie que le code passé dans ce paramètre existe dans la table du référentiel d'Apogée correspondante. Si ce code n'existe pas ou est lié à une donnée hors service, le Web Service remonte une erreur précise exprimant le fait que le code est invalide.

- Paramètre Pays annuelle

Ce paramètre est obligatoire. S'il n'est pas présent ou vide, le Web Service remonte une erreur précise exprimant le fait que ce paramètre est obligatoire.

Le Web Service vérifie que le code passé dans ce paramètre existe dans la table du référentiel d'Apogée correspondante. Si ce code n'existe pas ou est lié à une donnée hors service, le Web Service remonte une erreur précise exprimant le fait que le code est invalide.

- Paramètres Commune fixe

Si le paramètre « Pays fixe » correspond à la France alors ce paramètre est obligatoire. S'il n'est pas présent ou vide, le Web Service remonte une erreur précise exprimant le fait que ce paramètre est obligatoire.

Si le paramètre « Pays fixe » correspond à la France alors il faudra également que le connecteur vérifie que le code postal et le code commune de ce paramètre existent dans les tables du référentiel d'Apogée correspondantes. Si ces codes ne sont pas cohérents ou sont liés à des données hors service, le Web Service remonte une erreur précise exprimant le fait que les codes sont invalides.

Si le paramètre « Pays fixe » ne correspond pas à la France alors, il faudra que ce paramètre soit absent ou vide. S’il n’est pas vide, le Web Service remonte une erreur exprimant le fait que ce paramètre doit être vide.

- Paramètres Commune annuelle

Si le paramètre « Pays annuelle » correspond à la France alors ce paramètre est obligatoire. S’il n’est pas présent ou vide, le Web Service remonte une erreur précise exprimant le fait que ce paramètre est obligatoire.

Si le paramètre « Pays annuelle » correspond à la France alors il faudra également que le connecteur vérifie que le code postal et le code commune de ce paramètre existent dans les tables du référentiel d’Apogée correspondantes. Si ces codes ne sont pas cohérents ou sont liés à des données hors service, le Web Service remonte une erreur précise exprimant le fait que les codes sont invalides.

Si le paramètre « Pays annuelle » ne correspond pas à la France alors, il faudra que ce paramètre soit absent ou vide. S’il n’est pas vide, le Web Service remonte une erreur exprimant le fait que ce paramètre doit être vide.

- Paramètres Acheminement étranger fixe

Si le paramètre « Pays fixe » ne correspond pas à la France alors ce paramètre est obligatoire. S’il n’est pas présent ou vide, le Web Service remonte une erreur précise exprimant le fait que ce paramètre est obligatoire.

Si le paramètre « Pays fixe » correspond à la France alors ce paramètre devra être vide ou absent. Si il n’est pas vide, le Web Service remonte une erreur exprimant le fait que ce paramètre doit être vide.

- Paramètres Acheminement étranger annuelle

Si le paramètre « Pays annuelle » ne correspond pas à la France alors ce paramètre est obligatoire. S’il n’est pas présent ou vide, le Web Service remonte une erreur précise exprimant le fait que ce paramètre est obligatoire.

Si le paramètre « Pays annuelle » correspond à la France alors ce paramètre devra être vide ou absent. Si il n’est pas vide, le Web Service remonte une erreur exprimant le fait que ce paramètre doit être vide.

- Mise à jour des adresses

Si l’ensemble des conditions ci-dessus est respecté alors, le connecteur met à jours l’ensemble des données des deux adresses (fixe et annuelle).

Il y aura donc une modification de toutes les données des deux adresses dans Apogée.

2.2.4.6. Erreurs / Exceptions

En complément aux erreurs/exceptions ci-dessous, il faut ajouter les erreurs/exceptions du chapitre 2.1.6 ainsi que celles qui sont communes à l’ensemble des Web Services (voir § 3).

| Erreur | Type exception | Description |
|--------------------------|--|---|
| L’étudiant n’existe pas | technical.parameter.noncoherentinput.invalidUser | L’étudiant n’a pas été trouvé dans la base de données Apogée |
| Adresse fixe obligatoire | technical.parameter.nonpresentinput.adresseFixe | L’adresse fixe est obligatoire pour effectuer une modification des adresses |

| | | |
|---------------------------------|---|--|
| Adresse annuelle obligatoire | technical.parameter.nonpresentinput.adresseAnnuelle | L'adresse annuelle est obligatoire pour effectuer une modification des adresses |
| Type d'hébergement obligatoire | technical.parameter.nonpresentinput.typeHebergement | Le type d'hébergement pour l'adresse annuelle est obligatoire |
| Adresse1 fixe obligatoire | technical.parameter.nonpresentinput.libAd1 | La ligne 1 d'adresse fixe est obligatoire pour une modification des adresses |
| Adresse1 annuelle obligatoire | technical.parameter.nonpresentinput.libAd1 | La ligne 1 d'adresse annuelle est obligatoire pour une modification des adresses |
| Pays fixe obligatoire | technical.parameter.nonpresentinput.pays | Le pays de l'adresse fixe est obligatoire pour une modification des adresses |
| Pays annuelle obligatoire | technical.parameter.nonpresentinput.pays | Le pays de l'adresse annuelle est obligatoire pour une modification des adresses |
| Pays fixe invalide | technical.parameter.noncoherentinput.pays | Le pays de l'adresse fixe n'existe pas dans Apogée ou qu'il est hors service |
| Pays annuelle invalide | technical.parameter.noncoherentinput.pays | Le pays de l'adresse annuelle n'existe pas dans Apogée ou qu'il est hors service |
| Commune fixe obligatoire | technical.parameter.nonpresentinput.commune | La commune de l'adresse fixe est obligatoire car le pays de l'adresse fixe est la France |
| Commune annuelle obligatoire | technical.parameter.nonpresentinput.commune | La commune de l'adresse annuelle est obligatoire car le pays de l'adresse annuelle est la France |
| Commune fixe doit être vide | technical.parameter.nonnullinput.commune | La commune de l'adresse fixe doit être absente ou vide car le pays de l'adresse fixe n'est pas la France |
| Commune annuelle doit être vide | technical.parameter.nonnullinput.commune | La commune de l'adresse annuelle doit être absente ou vide le pays de l'adresse annuelle n'est pas la France |
| Commune fixe invalide | technical.parameter.noncoherentinput.codepostal-commune | La commune (code postal et code commune) de l'adresse fixe n'est pas valide : incohérence code postal et code commune, |

| | | |
|---|--|--|
| | | code postal ou code commune inexistant, code postal ou code commune hors service. |
| Commune annuelle invalide | <code>technical.parameter.noncoherentinput.codepostal-commune</code> | La commune (code postal et code commune) de l'adresse annuelle n'est pas valide : incohérence code postal et code commune, code postal ou code commune inexistant, code postal ou code commune hors service. |
| Acheminement étranger fixe obligatoire | <code>technical.parameter.nonpresentinput.libade</code> | Signifie que, vu que le pays de l'adresse fixe n'est pas la France, l'acheminement étranger de l'adresse fixe est obligatoire |
| Acheminement étranger annuelle obligatoire | <code>technical.parameter.nonpresentinput.libade</code> | L'acheminement étranger de l'adresse annuelle est obligatoire car le pays de l'adresse annuelle n'est pas la France |
| Acheminement étranger fixe doit être vide | <code>technical.parameter.nonnullinput.libade</code> | L'acheminement étranger de l'adresse fixe doit être absent ou vide car le pays de l'adresse fixe est la France |
| Acheminement étranger annuelle doit être vide | <code>technical.parameter.nonnullinput.libade</code> | L'acheminement étranger de l'adresse annuelle doit être absent ou vide car le pays de l'adresse annuelle est la France |

2.2.5. RECUPERATION DES INFORMATIONS ADMINISTRATIVES GENERALES DE L'ETUDIANT

Cette opération est liée à la méthode `recupererInfosAdmEtu` de la classe `EtudiantMetierServiceInterface` du stub client d'appel fournit par l'AMUE. La documentation de cette classe Java est disponible sous forme de JavaDoc.

2.2.5.1. Description

Cette opération doit permettre de récupérer les informations administratives générales de l'étudiant.

2.2.5.2. Version de l'opération

| Version | Principales évolutions | Date |
|---------|------------------------|----------|
| 0.1 | | 11/05/07 |

2.2.5.3. Paramètres en entrée

| Nom de la donnée | Nom technique | Type | Obligatoire |
|------------------|---------------|------|-------------|
|------------------|---------------|------|-------------|

| | | | |
|---------------|---------|--------|---|
| Code étudiant | _codEtu | String | O |
|---------------|---------|--------|---|

2.2.5.4. Données en sortie

Ce chapitre décrit de façon détaillée les données en sortie sous la forme d'un tableau :

| Nom de la donnée | Description | Nom technique | Type | Obligatoire |
|----------------------------------|------------------------------------|------------------------|---------------------|-------------|
| Infos administratives | Infos administratives générales | - | InfoAdmEtuDTO | N |
| Numéro étudiant | Numéro étudiant | numEtu | Integer | O |
| INE | Identifiant national de l'étudiant | numeroINE | String | O |
| Clé INE | Clé INE | cleINE | String | O |
| Nom | Nom patronymique | nomPatronymique | String | N |
| Nom usuel | Nom usuel | nomUsuel | String | O |
| Prénom 1 | Premier prénom | prenom1 | String | O |
| Prénom 2 | Deuxième prénom | prenom2 | String | N |
| Nationalité | Nationalité | nationaliteDTO | NationaliteDTO | O |
| Code nationalité | Code nationalité | codeNationalite | String | O |
| Libellé nationalité | Libellé nationalité | libNationalite | String | N |
| Date naissance | Date naissance | dateNaissance | Date | O |
| Témoin date de naissance estimée | Témoin date de naissance estimée | temoinDateNaissEstimee | String | N |
| Département de naissance | Département de naissance | departementNaissance | DepartementCourtDTO | N |
| Code département de naissance | Code département de naissance | codeDept | String | O |
| Libellé département de naissance | Libellé département de naissance | libDept | String | N |
| Pays de naissance | Pays de naissance | paysNaissance | PaysDTO | O |
| Code pays de naissance | Code pays de naissance | codPay | String | O |
| Libellé pays de naissance | Libellé pays de naissance | libPay | String | N |
| Libellé commune de naissance | Libellé commune de naissance | libVilleNaissance | String | N |
| Sexe | Sexe de l'étudiant | sexe | String | O |
| Situation familiale | Situation familiale | situationFamiliale | SitFamCourtDTO | N |
| Code | Code situation familiale | codeFam | String | O |
| Libellé | Libellé situation familiale | libFam | String | N |

| | | | | |
|---|--|-------------------------|---------------------|---|
| Libellé web | Libellé web situation familiale | libLongFam | String | N |
| Numéro boursier | Numéro boursier | numBoursier | String | N |
| Situation militaire | Situation militaire | situationMilitaire | SitMilCourtDTO | N |
| Code situation militaire | Code situation militaire | codeMil | String | |
| Libellé situation militaire | Libellé situation militaire | libSim | String | N |
| Libellé web situation militaire | Libellé web situation militaire | libMil | String | N |
| Témoin situation militaire en règle | Témoin situation militaire en règle (Oui/Non) | temoinSitMilEnRegle | String | N |
| Handicap de l'étudiant | Handicap de l'étudiant | handicap | TypHandicapCourtDTO | N |
| Code | Code handicap de l'étudiant | codeHandicap | String | O |
| Libellé | Libellé de l'étudiant | libThp | String | N |
| Libellé web | Libellé web de l'étudiant | codeHandicap | String | N |
| Année première inscription | Année première inscription dans l'enseignement supérieur | anneePremiereInscEnsSup | String | N |
| Année première inscription université | Année première inscription université | anneePremiereInscUniv | String | N |
| Année première inscription à l'étranger | Année première inscription à l'étranger | anneePremiereInscEtr | String | N |
| Lieu première inscription université | Lieu première inscription université | etbPremiereInscUniv | EtablissementDTO | N |
| Code établissement | Code établissement | codeEtb | String | O |
| Libellé établissement | Libellé établissement | libEtb | String | N |
| Année première inscription dans l'établissement | Année première inscription dans l'établissement | anneePremiereInscEtb | String | N |
| Bac | Bac | listeBacs | IndBacDTO | O |
| Code Bac | Code Bac | codBac | String | O |
| Libellé Bac | Libellé Bac | libelleBac | String | O |
| Année obtention bac | Année obtention bac | anneeObtentionBac | String | O |
| Mention bac | Mention bac | mentionBac | MentionNivBacDTO | N |
| Code | Code de la mention bac | codMention | String | O |

| | | | | |
|---|---|----------------|----------------------|---|
| Libellé | Libellé de la mention bac | libMention | String | N |
| Etablissement bac | Etablissement bac | etbBac | EtablissementDTO | N |
| Code | Code établissement bac | codeEtb | String | O |
| Libellé | Libellé établissement bac | libEtb | String | N |
| Type établissement bac | Type établissement bac | typeEtbBac | TypeEtablissementDTO | N |
| Code | Code type établissement bac | codTypeEtb | String | O |
| Libellé | Libellé type établissement bac | libLongTpe | String | N |
| Libellé web | Libellé web type établissement bac | libWebTpe | String | N |
| Département bac | Département bac | departementBac | DepartementCourtDTO | N |
| Code | Code département bac | codeDept | String | O |
| Libellé | Libellé département bac | libDept | String | N |
| Liste des interdits actuels de l'étudiant | Liste des interdits actuels de l'étudiant | listeBlocages | BlocageDTO[] | N |
| Code blocage | Code blocage | codBlocage | String | O |
| Libellé blocage | Libellé blocage | libBlocage | String | N |
| Type de blocage | Type de blocage | typBlocage | TypeBlocageDTO | O |
| Code | Code type de blocage | codTypBlocage | String | O |
| Libellé | Libellé type de blocage | libTypBlocage | String | N |
| Etape du blocage | Etape du blocage | etapeBlocage | EtapeDTO | N |
| Code | Code étape | codEtp | String | O |
| Libellé | Libellé étape | libEtp | String | N |

2.2.5.5. Règles de gestion

- Paramètre numéro étudiant

Il s'agit bien du COD_ETU présent dans Apogée.

Si l'étudiant n'existe pas, alors, le Web Service remonte une erreur précise exprimant le fait que l'étudiant n'existe pas.

- Lieu de naissance

Dans Apogée, il est enregistré soit le département de naissance (France), soit le pays de naissance. La différenciation se fait grâce au champ COD_TYP_DEP_PAY_NAI : « P » pour pays et « D » pour département.

Dans le cadre de ce connecteur, le choix qui a été fait est le suivant :

- Si il s'agit d'un pays (« P ») alors, le Web Service ne retourne que le pays dans les champs correspondants (code et libellé département).
- Si il s'agit d'un département (« D ») alors, le Web Service retourne le département dans les champs correspondants (code et libellé département) mais retournera également le pays France dans les champs correspondants (code et libellé pays).

Concernant la ville, dans la version 2007 d'Apogée il est prévu d'aller chercher la ville de naissance dans le table des communes afin d'avoir une meilleure qualité dans cette donnée. Le connecteur prend en compte cette évolution et il remonte donc la commune de naissance avec son code commune et son libellé commune.

Pour les villes qui seront toujours en saisie libre (pour les pays étrangers) le Web Service remonte uniquement le libellé commun.

- Sexe

Concernant le champ sexe de l'étudiant (F ou M), il n'y aura pas de transformation : le Web Service retourne F ou M.

- Témoin date de naissance estimée

Concernant le champ « Témoin date de naissance estimée » (O ou N), il n'y aura pas de transformation : le Web Service retourne O ou N.

- Liste des interdits

Le connecteur récupèrera la liste de tous les blocages que l'étudiant a au moment de l'appel au connecteur (date appel comprise entre la date début du blocage et la date de fin du blocage).

Si le blocage est lié à une étape particulière, elle sera remontée par le connecteur (code et libellé).

- Témoin situation militaire en règle (Oui / Non)

Concernant le champ « Témoin situation militaire en règle (Oui / Non) » (O ou N), il n'y aura pas de transformation : le Web Service retourne O ou N.

2.2.5.6. Erreurs / Exceptions

En complément aux erreurs/exceptions ci-dessous, il faut ajouter les erreurs/exceptions du chapitre 2.1.6 ainsi que celles qui sont communes à l'ensemble des WebServices (voir § 3).

| Erreur | Type exception | Description |
|-------------------------|--|--|
| L'étudiant n'existe pas | technical.parameter.noncoherentinput.invalidUser | L'étudiant n'a pas été trouvé dans la base de données Apogée |

2.2.6. RECUPERATION DU NUMERO DE SECURITE SOCIALE

Cette opération est liée à la méthode recupererNumeroSecuriteSociale de la classe EtudiantMetierServiceInterface du stub client d'appel fournit par l'AMUE. La documentation de cette classe Java est disponible sous forme de JavaDoc.

2.2.6.1. Description

Cette opération doit permettre de récupérer le numéro de sécurité sociale de l'étudiant.

2.2.6.2. Version de l'opération

| Version | Principales évolutions | Date |
|---------|------------------------|------------|
| 1 | | 04/06/2009 |

2.2.6.3. Paramètres en entrée

| Nom de la donnée | Nom technique | Type | Obligatoire |
|------------------|---------------|--------|-------------|
| Code étudiant | _codEtu | String | O |

2.2.6.4. Données en sortie

Ce chapitre décrit de façon détaillée les données en sortie sous la forme d'un tableau :

| Nom de la donnée | Description | Nom technique | Type | Obligatoire |
|--|--|---------------|------------|-------------|
| Numéro sécurité sociale | Numéro de sécurité sociale de l'étudiant | numSSoc | NumSSocDTO | O |
| Numéro de sécurité sociale de l'étudiant | Numéro de sécurité sociale de l'étudiant | numSSoc | String | O |
| Clé numéro sécurité sociale | Clé numéro de sécurité sociale | cleNumSSoc | String | N |

2.2.6.5. Règles de gestion

- Paramètre numéro étudiant

Il s'agit bien du COD_ETU présent dans Apogée.

Si l'étudiant n'existe pas, alors, le Web Service remonte une erreur précise exprimant le fait que l'étudiant n'existe pas.

- Numéro de sécurité sociale

Concernant le champ « numéro de sécurité sociale » et sa clé, il n'y aura pas de transformation : le Web Service retourne la valeur récupérée.

2.2.6.6. Erreurs / Exceptions

En complément aux erreurs/exceptions ci-dessous, il faut ajouter les erreurs/exceptions du chapitre 2.1.6 ainsi que celles qui sont communes à l'ensemble des WebServices (voir § 3).

| Erreur | Type exception | Description |
|---------------------------------------|---|---------------------------------------|
| Aucun étudiant n'a été trouvé | technical.security.invaliduser.user | Aucun étudiant n'a été trouvé |
| Etudiant non unique | technical.data.toomuchretrieve.etudiant | Etudiant non unique |
| Numéro de sécurité sociale non trouvé | technical.data.nullretrieve.numssoc | Numéro de sécurité sociale non trouvé |

2.2.7. RECUPERATION DES CALENDRIERS D'EXAMEN VALIDES D'UN ETUDIANT (VERSION 3)

Cette opération est liée à la méthode `recupererCalendriersExamenEtudiant_v3` de la classe `EtudiantMetierServiceInterface` du stub client d'appel fournit par l'AMUE. La documentation de cette classe Java est disponible sous forme de JavaDoc.

2.2.7.1. Description

Cette opération permet de récupérer l'ensemble des informations du calendrier d'examen d'un étudiant.

2.2.7.2. Version de l'opération

| Version | Principales évolutions | Date | Date fin de maintenance |
|---------|------------------------|------------|-------------------------|
| 1.0 | Création du connecteur | 31/05/2012 | |

2.2.7.3. Paramètres en entrée

| Nom de la donnée | Description | Nom technique | Type | Obligatoire |
|----------------------|-----------------------------|----------------------|--------|-------------|
| Année universitaire | Année universitaire épreuve | <code>_codAnu</code> | String | O |
| Numéro de l'étudiant | Code étudiant | <code>_codEtu</code> | String | O |
| Période d'examen | Code de la période d'examen | <code>_codPxa</code> | String | N |
| Calendrier d'examen | Code du calendrier d'examen | <code>_codCln</code> | String | N |

2.2.7.4. Données en sortie

Ce chapitre décrit de façon détaillée les données en sortie sous la forme d'un tableau :

| Nom de la donnée | Description | Nom technique | Type | Obligatoire |
|------------------------------------|-------------------|--------------------------------|------------------|-------------|
| Année universitaire épreuve | Année | <code>annee</code> | String | N |
| Etudiant | Etudiant | <code>etudiant</code> | EtudiantDTO4 | N |
| Code étudiant | Code | <code>codEtu</code> | String | N |
| Civilité | Civilité | <code>civilite</code> | String | N |
| Nom de l'étudiant | Nom | <code>nom</code> | String | N |
| Prénom de l'étudiant | Prénom | <code>prenom</code> | String | N |
| Date de naissance | Date de naissance | <code>dateNaissance</code> | String | N |
| Liste des périodes d'examen | Liste | <code>listPeriodeExamen</code> | PeriodeExaDTO4[] | N |
| Code période | Code | <code>codPeriode</code> | String | N |
| Libellé période | Libellé | <code>libPeriode</code> | String | N |
| Date de début | Date | <code>dateDebutPeriode</code> | String | N |
| Date de fin | Date | <code>dateFinPeriode</code> | String | N |

| | | | | |
|---|--------------|-----------------------|------------------------|---|
| Liste des calendriers d'examen | Liste | listCalendriersExamen | CalendrierExamenDTO3[] | N |
| Code calendrier | Code | codCalendrier | String | N |
| Libellé long calendrier | Libellé | libCalendrier | String | N |
| Liste des épreuves | Liste | listEpreuves | EpreuveDTO6[] | N |
| Code épreuve | Code | codEpreuve | String | N |
| Libellé court épreuve | Libellé | libCourtEpreuve | String | N |
| Libellé long épreuve | Libellé | libLongEpreuve | String | N |
| Type d'exercice | Type | typeExercice | String | N |
| Description des documents autorisés | Description | descDocsAuto | String | N |
| Durée | Durée | duree | String | N |
| Liste des caractéristiques techniques de l'épreuve | Liste | listCaracTech | CaracTechDTO[] | N |
| Code caractéristique technique de l'épreuve | Code | code | String | N |
| Libellé caractéristique technique de l'épreuve | Libellé | libelle | String | N |
| Liste des enseignants responsables | Liste | listEnsResp | PersonnelDTO3[] | N |
| Civilité | Civilité | civilite | String | N |
| Code enseignant | Code | code | String | N |
| Nom | Nom | nom | String | N |
| Prénom | Prénom | prenom | String | N |
| Liste des salles | Liste | listSalles | SalleDTO3[] | N |
| Code Salle | Code | codSalle | String | N |
| Libellé de la salle | Libellé | libSalle | String | N |
| Date d'examen | Date | dateExamen | String | N |
| Heure de début | Heure | heureDebut | String | N |
| Numéro de place étudiant | Place | place | String | N |
| Liste des caractéristiques techniques de la salle | Liste | listCaracTechSalle | CaracTechDTO[] | N |
| Code caractéristique technique de la salle | Code | code | String | N |
| Libellé caractéristique technique de la salle | Libellé | libelle | String | N |
| Code bâtiment | Code | codBat | String | N |
| Libellé | Libellé | libBat | String | N |
| Localisation | Localisation | localisationBat | String | N |
| Code type de salle | Code | codTypeSalle | String | N |
| Libellé du type de salle | Libellé | libType | String | N |
| Liste des surveillants affectés | Liste | listEnsAff | PersonnelDTO3[] | N |

| | | | | |
|-----------------|----------|----------|--------|---|
| Civilité | Civilité | civilite | String | N |
| Code enseignant | Code | code | String | N |
| Nom | Nom | nom | String | N |
| Prénom | Prénom | prenom | String | N |

2.2.7.5. Règles de gestion

- Année universitaire épreuve

Ce paramètre est obligatoire. Le format est AAAA. Il permet de filtrer la liste d'épreuves ramenées par le connecteur et de n'afficher que les informations des épreuves qui sont organisées dans le cadre de périodes d'examens au titre d'une année universitaire.

- Numéro de l'étudiant

Ce paramètre est obligatoire. Il permet de préciser pour quel étudiant le connecteur doit ramener les informations des calendriers d'examen valides.

- Code période d'examen

Ce paramètre est facultatif. Lorsqu'une période d'examen est saisie, le connecteur ne ramène que les informations des calendriers d'examen valides liés à la période saisie en entrée.

Si la période d'examen saisie n'est pas en service alors un message d'erreur sera remonté par le connecteur.

- Calendrier d'examen

Ce paramètre est facultatif. Lorsqu'un calendrier d'examen est saisi, le connecteur ne ramène que les informations du calendrier saisi en entrée.

Si aucun calendrier d'examen n'est saisi alors le connecteur ramènera tous les calendriers d'examen valides de l'étudiant.

Le calendrier d'examen saisi en paramètre doit être valide, c'est-à-dire que le tableau des incompatibilités a été généré.

- Heure de début d'examen

Le paramètre de sortie « heure de début d'examen » sera constitué de la concaténation de l'heure (DHH_DEB_PES) et des minutes (DMM_DEB_PES) de début d'examen. Ces deux informations seront séparées par le caractère « h ».

Exemple : 08h30

2.2.7.6. Erreurs / Exceptions

En complément aux erreurs/exceptions ci-dessous, il faut ajouter les erreurs/exceptions du chapitre 2.1.6 ainsi que celles qui sont communes à l'ensemble des WebServices (voir § 3).

| Erreur | Type exception | Description |
|--------------------------|--|---|
| L'année est obligatoire. | technical.webparameter.nonpresentinput .codanuobligatoire | Signifie que le paramètre année n'a pas été renseigné |

| | | |
|---|---|--|
| Le format de l'année est incorrect (AAAA). | technical.webparameter.noncoherentinput.codanuinvalid | Signifie que le format saisi pour l'année est incorrect. |
| Le code étudiant est obligatoire. | technical.webparameter.nonpresentinput.codetuobligatoire | Signifie que le paramètre code étudiant n'a pas été renseigné. |
| Aucun étudiant trouvé. | technical.webparameter.nullretrieve.codetuinexistent | Signifie que l'étudiant n'a été trouvé dans aucun calendrier de la période. |
| La période n'existe pas. | technical.webparameter.nullretrieve.codpxainexistent | Signifie que le code période renseigné n'a pas été trouvé dans la base de données Apogée. |
| Aucune période d'examen trouvée pour l'étudiant. | technical.webparameter.nullretrieve.aucuneperiode trouvee | Signifie qu'aucune période d'examen n'a été trouvée pour l'étudiant. |
| La période n'est pas en service | technical.webparameter.noncoherentinput.codpxahorsservice | Signifie que le code période renseigné n'est pas en service. |
| Le code calendrier n'existe pas. | technical.webparameter.nullretrieve.codclninexistent | Signifie que le code période calendrier renseigné n'a pas été trouvé dans la base de données Apogée. |
| Aucun calendrier d'examen trouvé pour l'étudiant. | technical.webparameter.nullretrieve.aucun calendrier trouve | Signifie qu'aucun calendrier d'examen n'a été trouvé pour l'étudiant. |
| Le calendrier n'est pas valide | technical.webparameter.noncoherentinput.codclninvalid | Signifie que le tableau des incompatibilités n'a pas été généré pour le code calendrier saisi. |

2.2.8. RECUPERATION DE LA LISTE DES ETUDIANTS

Cette opération est liée à la méthode `recupererListeEtudiants` de la classe `EtudiantMetierServiceInterface` du stub client d'appel fourni par l'AMUE. La documentation de cette classe Java est disponible sous forme de JavaDoc.

2.2.8.1. Description

Cette opération permet de récupérer une liste d'étudiants à partir de multiples paramètres.

2.2.8.2. Version de l'opération

| Version | Principales évolutions | Date |
|---------|------------------------|------------|
| 1.0 | Création du connecteur | 22/01/2010 |

2.2.8.3. Paramètres en entrée

| Nom de la donnée | Description | Nom technique | Type | Obligatoire |
|---|---|--------------------|---------------------------|-------------|
| Année demandée | Année universitaire demandée | annee | String | O |
| Liste diplôme | Code(s) diplôme | listDiplomes | EtudiantCritereListeDTO[] | O |
| Liste version diplôme (pour chaque diplôme) | Code(s) version diplôme | listVersion | String[] | O |
| Liste étape | Code(s) étape | listEtapes | EtudiantCritereListeDTO[] | O |
| Liste version étape (pour chaque étape) | Code(s) version étape | listVersion | String[] | O |
| Code collection de la VET | Code collection associée à la VET | codeCollectionVET | String | N |
| Code groupe VET | Code groupe pour une collection associée à la VET | codeGroupeVET | String | N |
| Liste éléments pédagogiques | Code(s) élément pédagogique | listELP | String[] | N |
| Code collection de l'ELP | Code collection associée à l'ELP | codeCollectionELP | String | N |
| Code groupe ELP | Code groupe pour une collection associée à l'ELP | codeGroupeELP | String | N |
| Liste code épreuve | Code(s) épreuve | listEPR | String | N |
| Liste type handicap | Code(s) type handicap | listTypeHandicap | String[] | N |
| Liste sportif de haut niveau | Code(s) sportif de haut niveau | listSportifHautNiv | String[] | N |
| Code régime inscription étape | Code régime inscription étape | codeRegimeInscrETP | String | N |
| Liste situation sociale | Code(s) situation sociale | listSitSociale | String[] | N |

| | | | | |
|---|---|--------------------------|----------|---|
| Sens échange international | Code sens échange international : A / E | sensEchInter | String | N |
| Code programme d'échange international | Code programme d'échange international | codeEchInter | String | N |
| Liste pays d'échange international | Code(s) pays d'échange international | listPaysEchInter | String[] | N |
| Liste type autre établissement fréquenté | Code(s) type autre établissement fréquenté | listTypeEtabFreq | String[] | N |
| Maintien de l'inscription dans un autre établissement | Maintien de l'inscription dans un autre établissement : C / P | maintienInscrAutreEtab | String | N |
| Liste centre gestion | Code(s) centre gestion | listCentreGestion | String[] | N |
| Liste composante | Code(s) composante | listComposante | String[] | N |
| Liste centre d'inscription pédagogique | Code(s) centre d'inscription pédagogique | listCIP | String[] | N |
| Témoin validation des acquis de l'expérience | O / N | temValAcquisXP | String | N |
| Témoin partenariat international | O / N | temPartInternationa l | String | N |
| Etat de la thèse | Etat de la thèse : E / S | etatThese | String | N |
| Type de résultat | Type de résultat : 0 / 1 | typeResultat | String | N |
| Objet résultat | VDI, VET, ELP, EPR | objetResultat | String | N |

2.2.8.4. Données en sortie

Ce chapitre décrit de façon détaillée les données en sortie sous la forme d'un tableau :

| Nom de la donnée | Description | Nom technique | Type | Obligatoire |
|--------------------------------|-----------------------------------|---------------|----------------|-------------|
| Liste des étudiants | Listes des étudiants | | EtudiantDTO2[] | O |
| Code étudiant | Code de l'étudiant | codEtu | String | O |
| Numéro INE étudiant (avec clé) | Numéro INE avec clé de l'étudiant | numeroIne | String | O |
| Nom étudiant | Nom de l'étudiant | nom | String | O |
| Prénom étudiant | Prénom de l'étudiant | prenom | String | O |
| Date naissance étudiant | Date de naissance de l'étudiant | dateNaissance | String | O |

2.2.8.5. Règles de gestion

- Remarques générales
 - Règle générale relative aux IA

Quels que soient les paramètres renseignés, le connecteur initialise un filtre sur les étudiants ayant une inscription administrative annuelle (table INS_ADM_ANU) non annulée (ETA_IAA <> 'A'), non résiliée (ETA_IAA <> 'R').

De plus, les étudiants retournés sont ceux ayant pour l'année demandée :

- SOIT une inscription administrative étape (table INS_ADM_ETP) **en règle** sur l'une des VDI et l'une des VET passées en paramètre.
- SOIT une inscription pédagogique en dette (table INS_PEDAGOGI_ETP avec COD_TYP_IPE='D') avec sur l'une des VET passées en paramètre et telle que la **VET visée** est **en règle**.

La **VET visée** est la VET sur laquelle l'étudiant est inscrit (table INS_DETTE_ETP : la VET visée correspond aux champs COD_ETP_2 et COD_VRS_VET_2, la VET en dette correspond aux champs COD_ETP_1 et COD_VRS_VET_1).

- SOIT une inscription pédagogique en crédit (table INS_PEDAGOGI_ETP avec COD_TYP_IPE='C') avec sur l'une des VET passées en paramètre et telle que la **VET d'inscription** est **en règle**.

La **VET d'inscription** est la VET sur laquelle l'étudiant est inscrit (table INS_ANTICIPE_ETP : la VET d'inscription correspond aux champs COD_ETP_2 et COD_VRS_VET_2, la VET en crédit correspond aux champs COD_ETP_1 et COD_VRS_VET_1).

Une inscription administrative étape est **en règle** si elle est non annulée (ETA_IAE <> 'A'), non résiliée (ETA_IAE <> 'R') et payée (ETA_PMT_IAE = 'P').

Cependant si au moins un des paramètres suivants est renseigné et valide :

- Code régime inscription étape
- Liste centre gestion
- Témoin validation des acquis de l'expérience
- Témoin partenariat international

Alors les inscriptions pédagogiques en dette ou en crédit ne sont pas considérées.

- Règle générale relative aux IP

Si au moins un des paramètres suivants est renseigné et valide :

- Liste centre d'inscription pédagogique
- Liste éléments pédagogiques
- Liste code épreuve
- Code collection de la VET
- Code groupe VET

- Code collection de l'ELP
- Code groupe ELP
- Objet résultat renseigné à 'ELP' ou 'EPR'

alors un filtre est ajouté sur les inscriptions pédagogiques normales, en dette ou en crédit (table INS_PEDAGOGI_ETP) pour l'année demandée.

- o Règle générale relative aux résultats

Les résultats étudiés sont toute session confondue (unique, 1,2).

En présence de 2 sessions, la priorité est donnée au résultat délibéré de la session 2.

Les résultats d'admissibilité ne sont pas étudiés, ni les résultats dont l'avancement n'est pas terminé.

Les résultats étudiés sont ceux relatifs à l'objet résultat passé en paramètre.

- Paramètre Année demandée

Il s'agit du COD_ANU présent dans APOGEE.

Si ce paramètre n'est pas renseigné alors le connecteur retournera l'erreur « L'année demandée est obligatoire ».

Sinon si ce paramètre n'est pas au format AAAA alors le connecteur retournera l'erreur « Le format de l'année demandée n'est pas correct ».

Sinon si le code n'existe pas dans APOGEE (table ANNEE_UNI) alors le connecteur retournera l'erreur « L'année demandée n'existe pas ».

Lorsque ce paramètre est valide, un filtre est ajouté sur les inscriptions administratives de l'année demandée (table INS_ADM_ANU).

- Paramètre Liste diplôme

Ce paramètre est une liste de codes diplôme (COD_DIP présent dans APOGEE).

Si la liste est absente ou vide alors le connecteur retournera l'erreur « Le diplôme est obligatoire ».

Sinon si l'un des codes diplôme de la liste n'existe pas dans APOGEE (table DIPLOME) alors le connecteur retournera l'erreur « Le diplôme n'existe pas ».

- Paramètre Liste version de diplôme

Ce paramètre est une liste de codes version de diplôme (COD_VRS_VDI présent dans APOGEE). Cette liste est propre à chaque code diplôme de la liste des diplômes.

Si la liste est absente ou vide pour l'un des diplômes alors le connecteur retournera l'erreur « La version de diplôme est obligatoire ».

Sinon si l'un des codes version de diplôme de la liste n'existe pas dans APOGEE pour le diplôme (table VERSION_DIPLOME) alors le connecteur retournera l'erreur « La version de diplôme n'existe pas ».

- Paramètre Liste étape

Ce paramètre est une liste de codes étape (COD_ETP présent dans APOGEE).

Si la liste est absente ou vide alors le connecteur retournera l'erreur « L'étape est obligatoire ».

Sinon si l'un des codes étape de la liste n'existe pas dans APOGEE (table ETAPE) alors le connecteur retournera l'erreur « L'étape n'existe pas ».

- Paramètre Liste version d'étape

Ce paramètre est une liste de codes version d'étape (COD_VRS_ETP présent dans APOGEE). Cette liste est propre à chaque code étape de la liste des étapes.

Si la liste est absente ou vide pour l'une des étapes alors le connecteur retournera l'erreur « La version d'étape est obligatoire ».

Sinon si l'un des codes version d'étape de la liste n'existe pas dans APOGEE pour le diplôme (table VERSION_ETAPE) alors le connecteur retournera l'erreur « La version d'étape n'existe pas ».

- Paramètre Code collection de la VET

Il s'agit du COD_EXT_COL présent dans APOGEE.

Si ce paramètre est renseigné alors le paramètre code groupe VET devient obligatoire. Dans ce cas, si le paramètre code groupe VET n'est pas renseigné alors le connecteur retournera l'erreur « Le groupe VET est obligatoire ».

Si ce paramètre est renseigné alors l'existence de la valeur pour toutes les VET passées en paramètre est vérifiée dans APOGEE : la collection (table COLLECTION) doit être associée à toutes les VET (table GPE_OBJ filtrée sur le type d'objet VET).

Si le code renseigné ne répond pas à ces critères alors le connecteur retournera l'erreur « La collection de la VET n'existe pas pour toutes les VET ».

Lorsque ce paramètre est renseigné et valide, un filtre est ajouté sur les étudiants inscrits dans l'un des groupes (table IND_AFFECTE_GPE) de la collection (tables GROUPE et COLLECTION) pour l'année demandée.

Lorsque ce paramètre est valide, un filtre est ajouté sur les inscriptions pédagogiques de l'année (cf. § Règle générale relative aux IP).

- Paramètre Code groupe VET

Il s'agit du COD_EXT_GPE présent dans APOGEE.

Si ce paramètre est renseigné alors le paramètre code collection de la VET devient obligatoire. Dans ce cas, si le paramètre code collection de la VET n'est pas renseigné alors le connecteur retournera l'erreur « La collection de la VET est obligatoire ».

Si ce paramètre et le code collection de la VET sont renseignés alors l'existence du code groupe VET pour la collection de la VET est vérifiée dans APOGEE (table GROUPE). Si le groupe VET n'existe pas pour la collection de la VET alors le connecteur retournera l'erreur « Le groupe VET n'existe pas ».

Lorsque ce paramètre est renseigné et valide, un filtre est ajouté sur les étudiants inscrits dans le groupe (table IND_AFFECTE_GPE) pour l'année demandée.

Lorsque ce paramètre est valide, un filtre est ajouté sur les inscriptions pédagogiques de l'année (cf. § Règle générale relative aux IP).

- Paramètre Liste élément pédagogique

Ce paramètre est une liste de codes élément (COD_ELP présent dans APOGEE).

Si l'un des codes élément de la liste n'existe pas dans APOGEE (table ELEMENT_PEDAGOGI) alors le connecteur retournera l'erreur « L'élément pédagogique n'existe pas ».

Lorsque ce paramètre est renseigné et valide, le connecteur ajoute un filtre sur les étudiants inscrits pédagogiquement à l'élément pour l'année demandée (table IND_CONTRAT_ELP).

Lorsque ce paramètre est valide, un filtre est ajouté sur les inscriptions pédagogiques de l'année (cf. § Règle générale relative aux IP).

- Paramètre Code collection de l'ELP

Il s'agit du COD_EXT_COL présent dans APOGEE.

Si ce paramètre est renseigné alors le paramètre code groupe ELP devient obligatoire. Dans ce cas, si le paramètre code groupe ELP n'est pas renseigné alors le connecteur retournera l'erreur « Le groupe ELP est obligatoire ».

Si ce paramètre est renseigné alors le paramètre liste élément pédagogique devient obligatoire. Dans ce cas, si la liste élément pédagogique est absente ou vide alors le connecteur retournera l'erreur « L'élément pédagogique est obligatoire ».

Si ce paramètre est renseigné alors l'existence de la valeur pour tous les éléments passés en paramètre est vérifiée dans APOGEE : la collection (table COLLECTION) doit être associée à tous les éléments (table GPE_OBJ filtrée sur le type d'objet ELP).

Si le code renseigné ne répond pas à ces critères alors le connecteur retournera l'erreur « La collection de l'ELP n'existe pas pour tous les ELP ».

Lorsque ce paramètre est renseigné et valide un filtre est ajouté sur les étudiants inscrits dans l'un des groupes (table IND_AFFECTE_GPE) de la collection (tables GROUPE et COLLECTION) pour l'année demandée.

Lorsque ce paramètre est valide, un filtre est ajouté sur les inscriptions pédagogiques de l'année (cf. § Règle générale relative aux IP).

- Paramètre Code groupe ELP

Il s'agit du COD_EXT_GPE présent dans APOGEE.

Si ce paramètre est renseigné alors le paramètre code collection de l'ELP devient obligatoire. Dans ce cas, si le paramètre code collection de l'ELP n'est pas renseigné alors le connecteur retournera l'erreur « La collection de l'ELP est obligatoire ».

Si ce paramètre et le code collection de l'ELP sont renseignés alors l'existence du code groupe ELP pour la collection de l'ELP est vérifiée dans APOGEE (table GROUPE). Si le groupe ELP n'existe pas pour la collection de l'ELP alors le connecteur retournera l'erreur « Le groupe ELP n'existe pas ».

Lorsque ce paramètre est renseigné et valide, un filtre est ajouté sur les étudiants inscrits dans le groupe (table IND_AFFECTE_GPE) pour l'année demandée.

Lorsque ce paramètre est valide, un filtre est ajouté sur les inscriptions pédagogiques de l'année (cf. § Règle générale relative aux IP).

- Paramètre Liste code épreuve

Ce paramètre est une liste de codes épreuve (COD_EPR présent dans APOGEE).

Si l'un des codes épreuve de la liste n'existe pas dans APOGEE (table EPREUVE) alors le connecteur retournera l'erreur « L'épreuve n'existe pas ».

Lorsque ce paramètre est renseigné et valide, un filtre est ajouté sur les étudiants ayant une ligne de résultat pour une des épreuves de la liste et pour l'année demandée (table RESULTAT_EPR).

Lorsque ce paramètre est valide, un filtre est ajouté sur les inscriptions pédagogiques de l'année (cf. § Règle générale relative aux IP).

- Paramètre Liste type handicap

Ce paramètre est une liste de types handicap (COD_THP présent dans APOGEE).

Si l'un des codes type handicap de la liste n'existe pas dans APOGEE (table TYPE_HANDICAP) alors le connecteur retournera l'erreur « Le type handicap n'existe pas ».

Lorsque ce paramètre est renseigné et valide, un filtre est ajouté sur les étudiants ayant l'un des types handicap passés en paramètre (table INDIVIDU).

- Paramètre Liste sportif de haut niveau

Ce paramètre est une liste de code sportif de haut niveau (COD_SPO présent dans APOGEE).

Si l'un des codes sportif de haut niveau de la liste n'existe pas dans APOGEE (table SPO_HAU_NIV) alors le connecteur retournera l'erreur « Le code sportif de haut niveau n'existe pas ».

Lorsque ce paramètre est renseigné et valide, un filtre est ajouté sur les étudiants ayant l'un des codes sportif de haut niveau passés en paramètre (table INS_ADM_ANU).

- Paramètre Code régime inscription étape

Il s'agit du COD_RGE présent dans APOGEE.

Si le code passé en paramètre n'existe pas dans APOGEE (table REGIME_INS) alors le connecteur retournera l'erreur « Le code régime inscription étape n'existe pas ».

Lorsque ce paramètre est renseigné et valide, un filtre est ajouté sur les étudiants ayant le code régime inscription étape passé en paramètre (table INS_ADM_ETP).

- Paramètre Liste situation sociale

Ce paramètre est une liste de code situation sociale (COD_SOC présent dans APOGEE).

Si l'un des codes situation sociale de la liste n'existe pas dans APOGEE (table SIT_SOCIALE) alors le connecteur retournera l'erreur « La situation sociale n'existe pas ».

Lorsque ce paramètre est renseigné et valide, un filtre est ajouté sur les étudiants ayant l'un des codes situation sociale passés en paramètre (table INS_ADM_ANU).

- Paramètre Code sens échange international

Il s'agit du TEM_SNS_PRG présent dans APOGEE.

Ce paramètre peut uniquement prendre les valeurs 'A' pour 'Accueil' et 'E' pour 'Envoi' sinon le connecteur retournera l'erreur « Le sens échange international n'existe pas ».

Lorsque ce paramètre est renseigné et valide, un filtre est ajouté sur les étudiants ayant le code sens échange international passé en paramètre (table INS_ADM_ANU).

- Paramètre Code programme d'échange international

Il s'agit du COD_PRG présent dans APOGEE.

Si ce code n'existe pas dans APOGEE (table PRG_ECHANGE) alors le connecteur retournera l'erreur « Le programme d'échange international n'existe pas ».

Lorsque ce paramètre est renseigné et valide, un filtre est ajouté sur les étudiants ayant le code programme d'échange international passé en paramètre (table INS_ADM_ANU).

- Paramètre Liste pays d'échange international

Ce paramètre est une liste de code pays (COD_PAYS_PRG présent dans APOGEE).

Si ce paramètre est renseigné alors le paramètre Code programme d'échange international devient obligatoire. Dans ce cas, si le paramètre Code programme d'échange international n'est pas renseigné alors le connecteur retournera l'erreur « Le programme d'échange international est obligatoire ».

Si l'un des codes pays d'échange international de la liste n'existe pas dans APOGEE (table PAYS) alors le connecteur retournera l'erreur « Le pays d'échange international n'existe pas ».

Lorsque ce paramètre est renseigné et valide, un filtre est ajouté sur les étudiants ayant l'un des codes pays d'échange international passés en paramètre (table INS_ADM_ANU).

- Paramètre Liste type autre établissement fréquenté

Ce paramètre est une liste de code type autre établissement fréquenté (COD_TPE_PRL_CHG présent dans APOGEE).

Si l'un des codes type autre établissement fréquenté de la liste n'existe pas dans APOGEE (table TYP_ETB) alors le connecteur retournera l'erreur « Le type d'autre établissement fréquenté n'existe pas ».

Lorsque ce paramètre est renseigné et valide, un filtre est ajouté sur les étudiants ayant l'un des codes pays type autre établissement fréquenté passés en paramètre (table INS_ADM_ANU).

- Paramètre Maintien de l'inscription dans un autre établissement

Il s'agit du COD_PRL_CHG présent dans APOGEE.

Ce paramètre peut uniquement prendre les valeurs 'P' pour 'Parallèle' et 'C' pour 'Changement' sinon le connecteur retournera l'erreur « Le code maintien de l'inscription dans un autre établissement n'existe pas ».

Lorsque ce paramètre est renseigné et valide, un filtre est ajouté sur les étudiants ayant le code maintien de l'inscription dans un autre établissement passé en paramètre (table INS_ADM_ANU).

- Paramètre Liste centre de gestion

Ce paramètre est une liste de centre de gestion (COD_CGE présent dans APOGEE).

Si l'un des codes centre de gestion de la liste n'existe pas dans APOGEE (table CENTRE_GESTION) alors le connecteur retournera l'erreur « Le centre de gestion n'existe pas ».

Lorsque ce paramètre est renseigné et valide, un filtre est ajouté sur les étudiants inscrits sur l'un des codes centre de gestion passés en paramètre (table INS_ADM_ETP).

- Paramètre Liste composante

Ce paramètre est une liste de composantes (COD_CMP présent dans APOGEE).

Si l'un des codes composante de la liste n'existe pas dans APOGEE (table COMPOSANTE) alors le connecteur retournera l'erreur « La composante n'existe pas ».

Lorsque ce paramètre est renseigné et valide, un filtre est ajouté sur les étudiants inscrits sur l'un des codes composante passés en paramètre (table INS_ADM_ETP).

- Paramètre Liste centre d'inscription pédagogique

Ce paramètre est une liste de centre d'inscription pédagogique (COD_CIP présent dans APOGEE).

Si l'un des codes centre d'inscription pédagogique de la liste n'existe pas dans APOGEE (table CENTRE_INS_PED) alors le connecteur retournera l'erreur « Le centre d'inscription pédagogique n'existe pas ».

Lorsque ce paramètre est renseigné et valide, un filtre est ajouté sur les étudiants inscrits sur l'un des codes centre d'inscription pédagogique en paramètre (table IND_CONTRAT_ELP).

Lorsque ce paramètre est valide, un filtre est ajouté sur les inscriptions pédagogiques de l'année (cf. § Règle générale relative aux IP).

- Paramètre Témoin validation des acquis de l'expérience

Il s'agit du TEM_VAE_CII présent dans APOGEE.

Ce paramètre peut uniquement prendre les valeurs 'O' pour 'Oui' et 'N' pour 'Non' sinon le connecteur retournera l'erreur « La valeur du témoin validation des acquis de l'expérience n'existe pas ».

Lorsque ce paramètre est renseigné et valide, un filtre est ajouté sur les étudiants ayant le témoin validation des acquis de l'expérience passé en paramètre dans la caractérisation de leur inscription administrative étape pour l'année demandée (table CARACT_IAE_IND).

- Paramètre Témoin partenariat international

Il s'agit du TEM_PI_CII présent dans APOGEE.

Ce paramètre peut uniquement prendre les valeurs 'O' pour 'Oui' et 'N' pour 'Non' sinon le connecteur retournera l'erreur « La valeur du témoin partenariat international n'existe pas ».

Lorsque ce paramètre est renseigné et valide, un filtre est ajouté sur les étudiants ayant le témoin partenariat international passé en paramètre dans la caractérisation de leur inscription administrative étape pour l'année demandée (table CARACT_IAE_IND).

- Paramètre Etat de la thèse

Il s'agit du ETA_THS présent dans APOGEE.

Ce paramètre peut uniquement prendre les valeurs 'E' pour 'En cours' et 'S' pour 'Soutenu' sinon le connecteur retournera l'erreur « L'état thèse n'existe pas ».

Lorsque ce paramètre est renseigné et valide, un filtre est ajouté sur les étudiants ayant une thèse dans l'état passé en paramètre pour l'une des versions de diplômes passées en paramètre (table THSES_HDR_SOUT).

- Paramètre Objet résultat

Ce paramètre peut uniquement prendre les valeurs 'VDI', 'VET', 'ELP' et 'EPR' sinon le connecteur retournera l'erreur « L'objet résultat n'existe pas ».

Si ce paramètre est renseigné alors le paramètre Type de résultat devient obligatoire. Dans ce cas, si le paramètre Type de résultat n'est pas renseigné alors le connecteur retournera l'erreur « Le type de résultat est obligatoire ».

Si ce paramètre vaut 'ELP' alors le paramètre Liste éléments pédagogiques devient obligatoire. Dans ce cas, si le paramètre Liste éléments pédagogiques n'est pas renseigné alors le connecteur retournera l'erreur « L'élément pédagogique est obligatoire ».

Si ce paramètre vaut 'EPR' alors le paramètre Liste code épreuve devient obligatoire. Dans ce cas, si le paramètre Liste code épreuve n'est pas renseigné alors le connecteur retournera l'erreur « L'épreuve est obligatoire ».

- Paramètre Type de résultat

Il s'agit du COD_NEG_TRE présent dans APOGEE.

Ce paramètre peut uniquement prendre les valeurs '0' et '1' sinon le connecteur retournera l'erreur « Le type de résultat n'existe pas ».

Si ce paramètre est renseigné alors le paramètre Objet résultat devient obligatoire. Dans ce cas, si le paramètre Objet résultat n'est pas renseigné alors le connecteur retournera l'erreur « L'objet résultat est obligatoire ».

Lorsque ce paramètre est renseigné et valide :

- S'IL vaut 'EPR' alors un filtre est ajouté sur les étudiants ayant un résultat du type passé en paramètre pour l'une des épreuves passées en paramètre et l'année demandée (tables RESULTAT_EPR et TYP_RESULTAT). Les résultats étudiés sont uniquement les résultats où l'étudiant est concerné (TEM_IND_CRN_EPR='CS'), de type admission (COD_ADM=1) dont l'avancement de délibération est terminé (ETA_AVC_EPR à 'T' dans GRP_RESULTAT_EPR pour l'épreuve, la session, l'admission et l'année demandée).
- SINON S'IL vaut 'ELP' alors un filtre est ajouté sur les étudiants ayant un résultat du type passé en paramètre pour l'un des éléments pédagogiques passés en paramètre et l'année demandée (tables RESULTAT_ELP et TYP_RESULTAT). Les résultats étudiés sont uniquement les résultats où l'étudiant est concerné (TEM_IND_CRN_ELP='CS'), de type admission (COD_ADM=1) dont l'avancement de délibération est terminé (ETA_AVC_ELP à 'T' dans GRP_RESULTAT_ELP pour l'élément, la session, l'admission et l'année demandée).
- SINON S'IL vaut 'VET' un filtre est ajouté sur les étudiants ayant un résultat du type passé en paramètre pour l'une des VET passées en paramètre et l'année demandée (tables RESULTAT_VET et TYP_RESULTAT). Les résultats étudiés sont uniquement les résultats où l'étudiant est concerné (TEM_IND_CRN_VET='CS'), de type admission (COD_ADM=1) dont l'avancement de délibération est terminé (ETA_AVC_VET à 'T' dans GRP_RESULTAT_VET pour la version d'étape, la session, l'admission et l'année demandée).
- SINON S'IL vaut 'VDI' alors un filtre est ajouté sur les étudiants ayant un résultat du type passé en paramètre pour l'une des VDI passées en paramètre et l'année demandée (tables RESULTAT_VDI et TYP_RESULTAT). Les résultats étudiés sont uniquement les résultats où l'étudiant est concerné (TEM_IND_CRN_VDI='CS'), de type admission (COD_ADM=1) dont l'avancement de délibération est terminé (ETA_AVC_VDI à 'T' dans GRP_RESULTAT_VDI pour la version de diplôme, la session, l'admission et l'année demandée).

Lorsque ce paramètre est renseigné à 'ELP' ou 'EPR', un filtre est ajouté sur les inscriptions pédagogiques de l'année (cf. § Règle générale relative aux IP).

- Absence de données

Si aucun étudiant correspondant aux critères n'est trouvé dans Apogée, le connecteur retournera l'erreur « Aucun étudiant correspondant aux critères n'a été trouvé ».

2.2.8.6. Erreurs / Exceptions

En complément aux erreurs/exceptions ci-dessous, il faut ajouter les erreurs/exceptions du chapitre 2.1.6 ainsi que celles qui sont communes à l'ensemble des WebServices (voir § 3).

| Erreur | Type exception | Description |
|--|--|---|
| L'année demandée est obligatoire | technical.webparameter.nonpresentinput.anneeobligatoire | Signifie que le paramètre année demandée n'a pas été renseigné. |
| Le format de l'année demandée n'est pas correct | technical.webparameter.noncoherentinput.anneeinvalide | Signifie que le format de l'année demandée n'est pas de type AAAA. |
| L'année demandée n'existe pas | technical.webparameter.nullretrieve.anneeinexistant | Signifie que l'année n'a pas été créée dans Apogee. |
| Le diplôme est obligatoire | technical.webparameter.nonpresentinput.listdiplomeobligatoire | Signifie que la liste des diplômes est soit absente soit vide. |
| Le diplôme n'existe pas | technical.webparameter.nullretrieve.listdiplomeinexistant | Signifie qu'au moins un des codes diplôme passés en paramètre n'existe pas dans APOGEE. |
| La version de diplôme est obligatoire | technical.webparameter.nonpresentinput.listversiondiplomeobligatoire | Signifie qu'au moins une des listes de versions de diplôme de l'un des diplômes est absente ou vide. |
| La version de diplôme n'existe pas | technical.webparameter.nullretrieve.listversiondiplomeinexistant | Signifie que l'un des couples code diplôme / code version de diplôme n'existe pas dans APOGEE. |
| L'étape est obligatoire | technical.webparameter.nonpresentinput.listetapeobligatoire | Signifie que la liste des étapes est soit absente soit vide. |
| L'étape n'existe pas | technical.webparameter.nullretrieve.listetapeinexistant | Signifie qu'au moins un des codes étape passés en paramètre n'existe pas dans APOGEE. |
| La version d'étape est obligatoire | technical.webparameter.nonpresentinput.listversionetapeobligatoire | Signifie qu'au moins une des listes de versions d'étape de l'une des étapes est absente ou vide. |
| La version d'étape n'existe pas | technical.webparameter.nullretrieve.listversionetapeinexistant | Signifie que l'un des couples code étape / code version d'étape n'existe pas dans APOGEE. |
| La collection de la VET n'existe pas pour toutes les VET | technical.webparameter.nullretrieve.codcollectionvetinexistant | Signifie que la collection de la VET n'existe pas dans APOGEE pour toutes les VET passées en paramètre. |

| | | |
|---|--|---|
| La collection de la VET est obligatoire | technical.webparameter.nonpresentinput. codecollectionvetobligatoire | Signifie qu'un code groupe VET a été renseigné mais pas le code collection de la VET. |
| Le groupe VET n'existe pas | technical.webparameter.nullretrieve.cod egroupevetinexistant | Signifie que le code groupe VET renseigné n'existe pas pour le code collection de la VET. |
| Le groupe VET est obligatoire | technical.webparameter.nonpresentinput. .codegroupevetobligatoire | Signifie qu'un code collection de la VET a été renseigné mais pas le groupe VET. |
| L'élément pédagogique n'existe pas | technical.webparameter.nullretrieve.liste lpinexistant | Signifie qu'au moins un des codes élément passés en paramètre n'existe pas dans APOGEE. |
| L'élément pédagogique est obligatoire | technical.webparameter.nonpresentinput. .listelpobligatoire | Signifie qu'un code collection de l'ELP a été renseigné mais pas le code ELP ou bien que l'objet résultat à été renseigné à ELP mais que la liste des ELP est absente ou vide |
| La collection de l'ELP n'existe pas pour tous les ELP | technical.webparameter.nullretrieve.cod ecollectionelpinexistant | Signifie que la collection de l'ELP n'existe pas dans APOGEE pour tous les éléments passés en paramètre. |
| La collection de l'ELP est obligatoire | technical.webparameter.nonpresentinput. .codecollectionelpobligatoire | Signifie qu'un code groupe ELP a été renseigné mais pas le code collection de l'ELP. |
| Le groupe ELP n'existe pas | technical.webparameter.nullretrieve.cod egroupeelpinexistant | Signifie que le code groupe ELP renseigné n'existe pas pour le code collection de l'ELP. |
| Le groupe ELP est obligatoire | technical.webparameter.nonpresentinput. .codegroupeelpobligatoire | Signifie qu'un code collection de l'ELP a été renseigné mais pas le groupe ELP. |
| L'épreuve n'existe pas | technical.webparameter.nullretrieve.liste prinexistant | Signifie qu'au moins un des codes épreuve passés en paramètre n'existe pas dans APOGEE. |
| L'épreuve est obligatoire | technical.webparameter.nonpresentinput. .listeprobligatoire | Signifie que l'objet résultat à été renseigné à EPR mais que la liste des épreuves est absente ou vide |
| Le type handicap n'existe pas | technical.webparameter.nullretrieve.listt ypehandicapinexistant | Signifie qu'au moins un des types handicap passés en paramètre n'existe pas dans APOGEE. |
| Le code sportif de haut niveau n'existe pas | technical.webparameter.nullretrieve.lists portifhautnivinexistant | Signifie qu'au moins un des codes sportif de haut niveau passé en |

| | | |
|--|--|---|
| | | paramètre n'existe pas dans APOGEE. |
| Le code régime inscription étape n'existe pas | technical.webparameter.nullretrieve.coderegimeinscretpinexistant | Signifie que le code régime inscription étape passé en paramètre n'existe pas dans APOGEE. |
| La situation sociale n'existe pas | technical.webparameter.nullretrieve.listsitsocialeinexistant | Signifie qu'au moins un des codes situation sociale passés en paramètre n'existe pas dans APOGEE. |
| Le sens échange international n'existe pas | technical.webparameter.nullretrieve.sensechinterinexistant | Signifie que le code sens échange international passé en paramètre n'existe pas dans APOGEE. |
| Le programme d'échange international est obligatoire | technical.webparameter.nonpresentinput.codeechinterobligatoire | Signifie qu'un code pays d'échange international a été renseigné mais pas le programme d'échange international. |
| Le programme d'échange international n'existe pas | technical.webparameter.nullretrieve.codeechinterinexistant | Signifie que le code programme d'échange international passé en paramètre n'existe pas dans APOGEE. |
| Le pays d'échange international n'existe pas | technical.webparameter.nullretrieve.listpaysechinterinexistant | Signifie qu'au moins un des codes pays d'échange international passés en paramètre n'existe pas dans APOGEE. |
| Le type d'autre établissement fréquenté n'existe pas | technical.webparameter.nullretrieve.listtypeetabfreqinexistant | Signifie qu'au moins un des codes type autre établissement fréquenté passés en paramètre n'existe pas dans APOGEE. |
| Le code maintien de l'inscription dans un autre établissement n'existe pas | technical.webparameter.nullretrieve.maintieninscrautreetabinexistant | Signifie que le code maintien de l'inscription dans un autre établissement passé en paramètre n'existe pas dans APOGEE. |
| Le centre de gestion n'existe pas | technical.webparameter.nullretrieve.listcentregestioninexistant | Signifie que au moins un des codes centre de gestion passés en paramètre n'existe pas dans APOGEE. |
| La composante n'existe pas | technical.webparameter.nullretrieve.listcomposanteinexistant | Signifie que le code composante passé en paramètre n'existe pas dans APOGEE |
| Le centre d'inscription pédagogique n'existe pas | technical.webparameter.nullretrieve.listcentreipinexistant | Signifie que au moins un des codes centre d'inscription pédagogique |

| | | |
|--|--|---|
| | | passés en paramètre n'existe pas dans APOGEE. |
| La valeur du témoin validation des acquis de l'expérience n'existe pas | technical.webparameter.nullretrieve.temvalacquisxp̄inexistent | Signifie que le témoin validation des acquis de l'expérience passé en paramètre n'existe pas dans APOGEE. |
| La valeur du témoin partenariat international n'existe pas | technical.webparameter.nullretrieve.tempartinternationalinexistent | Signifie que le partenariat international passé en paramètre n'existe pas dans APOGEE. |
| L'état thèse n'existe pas | technical.webparameter.nullretrieve.etatt̄heseinexistent | Signifie que le code état thèse passé en paramètre n'existe pas dans APOGEE. |
| Le type de résultat n'existe pas | technical.webparameter.nullretrieve.typēresultatinexistent | Signifie que le type de résultat passé en paramètre n'existe pas dans APOGEE. |
| Le type de résultat est obligatoire | technical.webparameter.nonpresentinput̄.typēresultatobligatoire | Signifie que l'objet résultat a été renseigné mais par le type de résultat |
| L'objet résultat n'existe pas | technical.webparameter.nullretrieve.objētresultatinexistent | Signifie que l'objet résultat passé en paramètre est différent de VDI, VET, ELP ou EPR |
| L'objet résultat est obligatoire | technical.webparameter.nonpresentinput̄.objet̄resultatobligatoire | Signifie que le type résultat a été renseigné mais par l'objet de résultat |
| Aucun étudiant correspondant aux critères n'a été trouvé | technical.webparameter.nullretrieve.auc̄unetudiant | Signifie qu'il n'y a pas d'étudiant répondant aux critères dans APOGEE. |

2.3. CONTRAT WSDL

2.3.1. CONTRAT D'IMPLEMENTATION

WSDL d'implémentation du service Etudiant décrivant le type (document) et l'encodage (litteral) du Web Service ainsi que l'URL d'accès à celui-ci.



EtudiantMetier_1011
2011_Impl.wsdl

2.3.2. CONTRAT ABSTRAIT

WSDL logique du service Etudiant de description syntaxique des opérations (méthodes) supportées par le Web Service.



EtudiantMetier_1011
2011.wsdl

2.3.3. SCHEMAS XML DES DONNEES D'ENTREE/SORTIE



EtudiantMetier.zip

3.ELEMENTS COMMUNS

L'ensemble des éléments communs aux WebServices autour d'Apogée sont répertoriés dans un document externe : manuel_technique_apogee_webservices.doc

Vous devez le consulter car il contient des informations indispensables à l'utilisation de ce WebService.