

CCSD / CNRS

Les Web Services HAL

Spécifications techniques de l'utilisation des WebServices
sur l'archive ouverte HAL

V 3.0

Sommaire

Sommaire.....	2
Les Web Services HAL.....	3
1. LES REFERENTIELS DANS HAL.....	5
1.1. Les environnements.....	5
1.2. Les types de publication.....	6
1.3. Les métadonnées.....	7
1.4. Les domaines scientifiques.....	9
1.5. Les laboratoires.....	11
1.6. Les affiliations d'un laboratoire.....	15
1.7. Les revues.....	16
1.8. Les types de fichiers.....	19
1.9. Embargo sur les fichiers.....	20
1.10. Version des fichiers déposés.....	21
1.11. Les collections.....	22
1.12. Métadonnée audience.....	23
1.13. Les organismes de thèses.....	24
1.14. Les écoles doctorales.....	25
1.15. Les projets ANR.....	27
1.16. Les projets européens.....	28
2. Le dépôt.....	29
2.1. Dépôt d'un nouveau document dans l'archive.....	29
2.2. Correction d'un article.....	31
2.3. Dépôt d'une nouvelle version.....	32
2.4. Modification des métadonnées.....	34
2.5. Ajout / Modification du type de publication.....	35
2.6. Ajout de domaine scientifique.....	36
2.7. Ajout de fichier à une notice bibliographique.....	37
2.8. Ajout de fichiers annexes à un dépôt avec fichier.....	38
2.9. Ajout d'un article dans une collection.....	38
2.10. Statut d'un article.....	40
2.11. Mise en ligne des fichiers d'un article sous embargo.....	41
3. La recherche.....	42
3.1. Existence.....	42
3.2. Recherche.....	42
3.3. Récupération des métadonnées d'un article.....	44
3.4. Récupération du texte intégral d'un article.....	45
3.5. Recherche de thèses.....	46
3.6. Récupération des statistiques de consultation d'un article.....	47
4. Les doublons.....	48

Les Web Services HAL

L'archive ouverte scientifique **HAL** (<http://hal.archives-ouvertes.fr>) dispose de **Services Web** (WS) fondés sur le protocole **SOAP** utilisant le style **WSDL document/literal**. HAL définit ainsi un certain nombre de méthodes qui vont permettre à un système informatique local de réaliser des opérations similaires à celles que fait un contributeur lorsqu'il utilise l'application HAL au travers de son portail Web.

Les méthodes principales découlent directement d'un dépôt fournissant des données qualifiées à l'aide des référentiels obligatoires utilisés dans le modèle de données de HAL :

- ❖ Le WS permettant d'accéder en **consultation aux référentiels** internes de HAL (métadonnées, laboratoires, domaines scientifiques, etc.) ; accessible via le protocole HTTP à l'URL :
 - **Serveur de production** : <http://hal.archives-ouvertes.fr/ws/ref.php>
wSDL : <http://hal.archives-ouvertes.fr/ws/ref.php?wsdl>
 - **Serveur de développement / test** : <http://paol.ccsd.cnrs.fr/ws/ref.php>
wSDL : <http://paol.ccsd.cnrs.fr/ws/ref.php?wsdl>

- ❖ Le WS permettant de **déposer, compléter, modifier** des articles scientifiques ; accessible via le protocole HTTP à l'URL :
 - **Serveur de production** : <http://hal.archives-ouvertes.fr/ws/submit.php>
wSDL : <http://hal.archives-ouvertes.fr/ws/submit.php?wsdl>
 - **Serveur de développement / test** : <http://paol.ccsd.cnrs.fr/ws/submit.php>
wSDL : <http://paol.ccsd.cnrs.fr/ws/submit.php?wsdl>

- ❖ Le WS permettant de **consulter ou rechercher** les articles scientifiques déposés sur HAL ; accessible via le protocole HTTP à l'URL :
 - **Serveur de production** : <http://hal.archives-ouvertes.fr/ws/search.php>
wSDL : <http://hal.archives-ouvertes.fr/ws/search.php?wsdl>
 - **Serveur de développement / test** : <http://paol.ccsd.cnrs.fr/ws/search.php>
wSDL : <http://paol.ccsd.cnrs.fr/ws/search.php?wsdl>

Vous pouvez trouver des exemples d'implémentation (PHP) de ces Services Web sur le site <http://www.ccsd.cnrs.fr/support.html#clientws>

1. LES REFERENTIELS DANS HAL

1.1. Les environnements

Hal dispose de plusieurs portails d'entrée différents (environnement ou instance), avec des métadonnées spécifiques.

getRefInstance

Description

Récupération de l'ensemble des environnements de la plateforme HAL.

Liste des paramètres

aucun

Valeurs de retour

❖ **getRefInstanceResult** (*ArrayOfStrings*)

Liste contenant les noms des différents environnements disponibles de l'archive HAL

1.2. Les types de publication

HAL permet de qualifier un dépôt par un « type de publication » lorsque le papier est publié (article de journal, chapitre d'ouvrage, etc.)

getRefTypePubli

Description

Récupération de la liste des types de publications en fonction de l'environnement.

Liste des paramètres

❖ **instance** (*string*)

Nom de l'environnement pour lequel on souhaite récupérer la liste des types de publication.

La liste des environnements peut être récupérée par la méthode [getRefInstance](#)

❖ **lang** (*string*)

Code langue au format ISO 639-1 (alpha-2)

Valeurs de retour

❖ **getRefTypePubliResult** (*ArrayOfTypePubliStruct* : tableau de [TypePubliStruct](#))

Liste des types de publications. L'attribut *type* permet de savoir si le type de publication est déposable avec ou sans fichier.

1.3. Les métadonnées

Ces référentiels permettent de savoir quelles informations sont à fournir lors du dépôt d'un article dans l'archive HAL. La liste des métadonnées dépend de l'instance et du type de publication.

getRefMetaData

Description

Récupération de la liste des métadonnées à fournir lors du dépôt d'un article en fonction de l'instance et du type de publication.

Liste des paramètres

❖ **instance** (*string*)

Nom de l'environnement.

La liste des environnements peut être récupérée par la méthode [getRefInstance](#)

❖ **typePubli** (*string*)

Identifiant du type de publication.

La liste des types de publications peut être récupérée par la méthode [getRefTypePubli](#)

❖ **lang** (*string*)

code langue au format ISO 639-1 (alpha-2)

Valeurs de retour

❖ **getRefMetaDataResult** (*ArrayOfMetaDataStruct* : tableau de [MetaDataStruct](#))

Liste des métadonnées pour un type de publication donné ou accessibles à tous les types.

getRefMetaData_forTypePubli

Description

Cette méthode permet de récupérer les métadonnées spécifiques à fournir à HAL lors de l'ajout de la typologie de publication.

Liste des paramètres

- ❖ **instance** (*string*)
Nom de l'environnement.
La liste des environnements peut être récupérée par la méthode [getRefInstance](#)
- ❖ **typePubli** (*string*)
Identifiant du type de publication.
La liste des types de publications peut être récupérée par la méthode [getRefTypePubli](#)
- ❖ **lang** (*string*)
Code langue au format ISO 639-1 (alpha-2)

Valeurs de retour

- ❖ **getRefMetaData_forTypePubliResult** (*ArrayOfMetaDataStruct* : tableau de [MetaDataStruct](#))
Liste des métadonnées pour un type de publication donné ou accessibles à tous les types.

1.4. Les domaines scientifiques

Un article peut être déposé dans un ou plusieurs domaines. Ce référentiel contient la liste des domaines qui peuvent être choisis pour déposer un article suivant un portail.

getRefSubject_byCode

Description

Accès aux informations d'un domaine à partir de son code

Liste des paramètres

- ❖ **instance** (*string*)
Nom de l'environnement.
La liste des environnements peut être récupérée par la méthode [getRefInstance](#)
- ❖ **code** (*string*)
Identifiant du domaine scientifique.
La liste des codes peut être retournée par la méthode [getRefSubject](#)
- ❖ **lang** (*string*)
Code langue au format ISO 639-1 (alpha-2)

Valeurs de retour

- ❖ **getRefSubject_byCodeResult** ([SubjectStruct](#))
Informations sur un domaine scientifique.

getRefSubject

Description

Récupération du référentiel des domaines scientifiques pour une instance donnée.

Liste des paramètres

- ❖ **instance** (*string*)
Nom de l'environnement.
La liste des environnements peut être récupérée par la méthode [getRefInstance](#)

Valeurs de retour

- ❖ **getRefSubjectResult** (*ArrayOfStrings*)
Liste des codes des domaines du référentiel.

getRefSubject_byCodeParent

Description

Accès aux codes des sous domaines d'un domaine scientifique

Liste des paramètres

❖ **instance** (*string*)

Nom de l'environnement.

La liste des environnements peut être récupérée par la méthode [getRefInstance](#)

❖ **code** (*string*)

Identifiant du domaine scientifique.

La liste des codes peut être retournée par la méthode [getRefSubject](#)

Valeurs de retour

❖ **getRefSubject_byCodeParentResult** (*ArrayOfStrings*)

Liste des codes des sous domaines du domaine passé en argument

1.5. Les laboratoires

Ce référentiel liste les différents laboratoires connus de HAL. Les différentes méthodes permettant d'accéder au référentiel des laboratoires vont retourner un objet de type [LabStruct](#) ou un tableau d'identifiants de laboratoire. L'objet [LabStruct](#) contient toutes les informations définissant un laboratoire.

getRefLab_byId

Description

Accès aux informations d'un laboratoire à partir de son identifiant

Liste des paramètres

- ❖ **knownLabid** (*int*)
Identifiant d'un laboratoire dans l'archive HAL

Valeurs de retour

- ❖ **getRefLab_byIdResult** ([LabStruct](#))
Objet contenant les différentes informations sur un laboratoire

getRefLab

Description

Récupération du référentiel des laboratoires

Liste des paramètres

- ❖ **from** (*date au format AAAA-MM-JJ*)
Récupération des identifiants des laboratoires ajoutés ou modifiés depuis la date *from*. Si ce paramètre est vide, la totalité du référentiel sera retourné.

Valeurs de retour

- ❖ **getRefLabResult** (*ArrayOfInts*)
Liste des identifiants des laboratoires

getRefLab_byAffi

Description

Récupération du référentiel des laboratoires à partir d'une affiliation

Liste des paramètres

- ❖ **affi** (*string*)
Affiliation du laboratoire. Il est possible de rechercher un laboratoire CNRS par son code unité.
- ❖ **searchType** (*string*)
Type de recherche. Ce paramètre peut prendre les valeurs *like* (contient), *start_by* (commence par), *end_by* (fini par), *is_exactly* (correspond exactement à). Par défaut, la valeur est *like*.
- ❖ **from** (*date au format AAAA-MM-JJ*)
Récupération des identifiants des laboratoires ajoutés ou modifiés depuis la date *from*. Si ce paramètre est vide, la totalité du référentiel sera retourné.

Valeurs de retour

- ❖ **getRefLab_byAffiResult** (*ArrayOfInts*)
Liste des identifiants des laboratoires correspondant aux critères de la recherche

getRefLab_byShortName

Description

Récupération du référentiel des laboratoires à partir du sigle d'un laboratoire

Liste des paramètres

- ❖ **shortName** (*string*)
Sigle du laboratoire recherché
- ❖ **searchType** (*string*)
Type de recherche. Ce paramètre peut prendre les valeurs *like* (contient), *start_by* (commence par), *end_by* (fini par), *is_exactly* (correspond exactement à). Par défaut, la valeur est *like*.
- ❖ **from** (*date au format AAAA-MM-JJ*)
Récupération des identifiants des laboratoires ajoutés ou modifiés depuis la date *from*. Si ce paramètre est vide, la totalité du référentiel sera retourné.

Valeurs de retour

- ❖ **getRefLab_byShortNameResult** (*ArrayOfInts*)
Liste des identifiants des laboratoires correspondant aux critères de la recherche

getRefLab_byName

Description

Récupération du référentiel des laboratoires à partir du nom d'un laboratoire

Liste des paramètres

- ❖ **name** (*string*)
Nom du laboratoire recherché
- ❖ **searchType** (*string*)
Type de recherche. Ce paramètre peut prendre les valeurs *like* (contient), *start_by* (commence par), *end_by* (fini par), *is_exactly* (correspond exactement à). Par défaut, la valeur est *like*.
- ❖ **from** (*date au format AAAA-MM-JJ*)
Récupération des identifiants des laboratoires ajoutés ou modifiés depuis la date *from*. Si ce paramètre est vide, la totalité du référentiel sera retourné.

Valeurs de retour

- ❖ **getRefLab_byNameResult** (*ArrayOfInts*)
Liste des identifiants des laboratoires correspondant aux critères de la recherche

getRefLab_byAuthor

Description

Récupération des derniers laboratoires associés à un auteur lors du dernier dépôt

Liste des paramètres

- ❖ **lastname** (*string*)
Nom de l'auteur
- ❖ **firstname** (*string*)
Prénom de l'auteur
- ❖ **email** (*string*)
Mail de l'auteur (optionnel)

Valeurs de retour

- ❖ **getRefLab_byAuthorResult** (*ArrayOfInts*)
Liste des identifiants des laboratoires correspondant aux critères de la recherche

getRefLab_search

Description

Recherche dans le référentiel des laboratoires. Méthode retournant au maximum 15 objets laboratoires dont le sigle, le nom ou une des affiliations correspond à la chaîne de caractère passée en argument.

Liste des paramètres

- ❖ **search** (*string*)
Chaîne de caractère à rechercher

Valeurs de retour

- ❖ **getRefLab_searchResult** (*ArrayOfLabStruct* : tableau de [LabStruct](#))
Liste d'objets laboratoire correspondant aux critères de la recherche

addRefLab

Description

Ajout d'un nouveau laboratoire au référentiel de HAL. Ce nouveau laboratoire ne sera pas valide (ne sera pas retourné par les autres méthodes) mais possèdera un labid et pourra être utilisé lors d'un dépôt.

Liste des paramètres

- ❖ **identification** ([UserStruct](#))
Identifiants d'un compte sur la plateforme HAL
- ❖ **lab** (*addLabStruct*)
Laboratoire à rajouter

Valeurs de retour

- ❖ **addRefLabResult** (*Int*)
Identifiant du nouveau laboratoire

1.6. Les affiliations d'un laboratoire

Le référentiel des affiliations est constitué de tous les organismes qui peuvent être associés à un laboratoire. Les organismes peuvent être des institutions (CNRS, CEA, ...), des universités, des grandes écoles et des partenaires privés.

getRefAffi

Description

Récupération du référentiel des affiliations

Liste des paramètres

- ❖ **from** (*date au format AAAA-MM-JJ*)

Récupération des affiliations ajoutées ou modifiées depuis la date *from*. Si ce paramètre est vide, la totalité du référentiel sera retourné.

Valeurs de retour

- ❖ **getRefAffiResult** (*ArrayOfStrings*)

Liste des intitulés des affiliations du référentiel

getRefAffi_byName

Description

Accès aux informations d'un laboratoire à partir de son identifiant

Liste des paramètres

- ❖ **search** (*string*)

Nom du laboratoire recherché

- ❖ **searchType** (*string*)

type de recherche. Ce paramètre peut prendre les valeurs *like* (contient), *start_by* (commence par), *end_by* (fini par), *is_exactly* (correspond exactement à). Par défaut, la valeur est *like*.

- ❖ **from** (*date au format AAAA-MM-JJ*)

Récupération des affiliations ajoutées ou modifiées depuis la date *from*. Si ce paramètre est vide, la totalité du référentiel sera retourné.

Valeurs de retour

- ❖ **getRefAffi_byNameResult** (*ArrayOfStrings*)

Liste des intitulés des affiliations du référentiel

1.7. Les revues

Les méthodes suivantes retournent soit un objet de type [ReviewStruct](#), soit un tableau d'entier pointant vers un objet revue. Cet objet contient le nom, le sigle, l'ISSN, l'éditeur et l'EISSN de la revue.

getRefReview_byId

Description

Récupération des informations d'une revue

Liste des paramètres

- ❖ **id** (*int*)
Identifiant unique d'une revue sur l'archive HAL

Valeurs de retour

- ❖ **getRefReview_byIdResult** ([ReviewStruct](#))
Objet contenant les différentes informations sur une revue

getRefReview

Description

Récupération du référentiel des revues

Liste des paramètres

- ❖ **from** (*date au format AAAA-MM-JJ*)
Récupération des affiliations ajoutées ou modifiées depuis la date *from*. Si ce paramètre est vide, la totalité du référentiel sera retourné

Valeur de retour

- ❖ **getRefReviewResult** (*ArrayOfInts*)
Tableau contenant les identifiants HAL des revues du référentiel.

getRefReview_byName

Description

Recherche dans le référentiel à partir du nom de la revue

Liste des paramètres

- ❖ **name** (*string*)
Nom de la revue recherchée
- ❖ **searchType** (*string*)
Type de recherche. Ce paramètre peut prendre les valeurs *like* (contient), *start_by* (commence par), *end_by* (fini par), *is_exactly* (correspond exactement à). Par défaut, la valeur est *like*.

Valeurs de retour

- ❖ **getRefReview_byNameResult** (*ArrayOfInts*)
Tableau contenant les identifiants HAL des revues du référentiel.

getRefReview_byEditor

Description

Recherche dans le référentiel à partir du nom de l'éditeur

Liste des paramètres

- ❖ **editor** (*string*)
Nom de l'éditeur recherché
- ❖ **searchType** (*string*)
Type de recherche. Ce paramètre peut prendre les valeurs *like* (contient), *start_by* (commence par), *end_by* (fini par), *is_exactly* (correspond exactement à). Par défaut, la valeur est *like*.

Valeurs de retour

- ❖ **getRefReview_byEditorResult** (*ArrayOfInts*)
Tableau contenant les identifiants HAL des revues du référentiel.

getRefReview_byISSN

Description

Recherche dans le référentiel à partir de l'ISSN de la revue

Liste des paramètres

❖ **issn** (*string*)

ISSN de la revue recherchée

❖ **searchType** (*string*)

Type de recherche. Ce paramètre peut prendre les valeurs *like* (contient), *start_by* (commence par), *end_by* (fini par), *is_exactly* (correspond exactement à). Par défaut, la valeur est *like*.

Valeurs de retour

❖ **getRefReview_byISSNResult** (*ArrayOfInts*)

Tableau contenant les identifiants HAL des revues du référentiel.

1.8. Les types de fichiers

Référentiel contenant tous les formats de fichiers déposables sur HAL.

getRefFormat

Description

Retourne la liste des formats de fichiers déposables sur HAL.

Liste des paramètres

❖ **instance** (*string*)

Nom de l'environnement.

La liste des environnements peut être récupérée par la méthode [getRefInstance](#)

❖ **lang** (*string*)

Code langue au format ISO 639-1 (alpha-2)

Valeurs de retour

❖ **getRefFormatResult** (*ArrayOfFormatStruct* tableau de [FormatStruct](#))

Tableau d'objet contenant les informations sur les formats de fichiers déposables dans une instance de HAL.

1.9. Embargo sur les fichiers

Lors du dépôt d'un article, l'auteur peut vouloir définir une période d'embargo pendant laquelle le document ne sera pas visible.

getRefDateVisible

Description

Retourne les différentes durées d'embargo disponibles.

Liste des paramètres

- ❖ **aucun**

Valeurs de retour

- ❖ **getRefDateVisibleResult** (*ArrayOfDateVisibleStruct* tableau de [DateVisibleStruct](#))
Tableau d'objet contenant les différentes durées d'embargo disponibles.

1.10. Version des fichiers déposés

Lors d'un dépôt sur HAL d'un article scientifique avec texte attaché, il est obligatoire de donner le droit sur les fichiers déposés (ex. fichier auteur, fichier éditeur). Cette méthode permet de connaître les différentes valeurs possibles.

getRefRightFulltext

Description

Retourne la liste des droits sur les fichiers

Liste des paramètres

- ❖ **aucun**

Valeurs de retour

- ❖ **getRefRightFulltextResult** (*ArrayOfRightFulltextStruct* tableau de [RightFulltextStruct](#))
Tableau d'objet contenant les différents droits disponibles.

1.11. Les collections

HAL gère des collections au dessus des dépôts. Cette méthode permet de connaître la liste des collections administrée par un utilisateur.

getRefCollection

Description

Retourne la liste des tampons d'un utilisateur sur l'archive HAL

Liste des paramètres

- ❖ **instance** (*string*)
Nom de l'environnement.
La liste des environnements peut être récupérée par la méthode [getRefInstance](#)
- ❖ **uid** (*string*)
Login de l'utilisateur dans l'archive HAL

Valeurs de retour

- ❖ **getRefCollectionResult** (*ArrayOfCollectionStruct* tableau de [CollectionStruct](#))
Tableau d'objet contenant les informations sur les collections d'un utilisateur.

1.12. Métadonnée audience

Référentiel contenant les différentes valeurs pour la métadonnée audience de HAL.

getRefAudience

Description

Retourne la liste des valeurs pour la métadonnée « audience »

Liste des paramètres

- ❖ **lang** (*string*)
Code langue au format ISO 639-1 (alpha-2)

Valeurs de retour

- ❖ **getRefAudienceResult** (*ArrayOfAudienceStruct* tableau de [AudienceStruct](#))
Tableau d'objet contenant les informations sur les valeurs.

1.13. Les organismes de thèses

Ce référentiel ne s'utilise que pour les dépôts de thèses ou de HDR (sur l'instance TEL)

getRefOrgThe

Description

Retourne la liste des organismes de thèses du référentiel

Liste des paramètres

❖ **pays** (*string*)

Identifiant du pays (code ISO 3166-1). Par défaut, et si le paramètre est vide, tous les organismes sont retournés.

Valeurs de retour

❖ **getRefOrgTheResult** (*ArrayOfOrgTheStruct* tableau de [OrgTheStruct](#))

Tableau d'objet contenant les informations sur les organismes de thèses.

getRefOrgThe_byName

Description

Recherche dans le référentiel à partir du nom de l'organisme

Liste des paramètres

❖ **name** (*string*)

Nom de l'organisme

❖ **searchType** (*string*)

Type de recherche. Ce paramètre peut prendre les valeurs *like* (contient), *start_by* (commence par), *end_by* (fini par), *is_exactly* (correspond exactement à). Par défaut, la valeur est *like*.

Valeurs de retour

❖ **getRefOrgThe_byNameResult** (*ArrayOfOrgTheStruct* tableau de [OrgTheStruct](#))

Tableau d'objet contenant les informations sur les organismes de thèses.

1.14. Les écoles doctorales

Ce référentiel ne s'utilise que pour les dépôts de thèses ou de HDR (sur l'instance TEL)

getRefSchool

Description

Retourne le référentiel des écoles doctorales

Liste des paramètres

- ❖ **aucun**

Valeurs de retour

- ❖ **getRefSchoolResult** (*ArrayOfSchoolStruct* tableau de [SchoolStruct](#))
Tableau d'objet contenant les informations sur les écoles doctorales

getRefSchool_byName

Description

Recherche dans le référentiel à partir du nom de l'école doctorale

Liste des paramètres

- ❖ **name** (*string*)
Nom de l'école
- ❖ **searchType** (*string*)
Type de recherche. Ce paramètre peut prendre les valeurs *like* (contient), *start_by* (commence par), *end_by* (fini par), *is_exactly* (correspond exactement à). Par défaut, la valeur est *like*.

Valeurs de retour

- ❖ **getRefSchool_byNameResult** (*ArrayOfSchoolStruct* tableau de [SchoolStruct](#))
Tableau d'objet contenant les informations sur les écoles doctorales

getRefSchool_byOrgThe

Description

Recherche dans le référentiel à partir de son organisme de thèse

Liste des paramètres

❖ **name** (*string*)

Nom de l'organisme de thèse

❖ **searchType** (*string*)

Type de recherche. Ce paramètre peut prendre les valeurs *like* (contient), *start_by* (commence par), *end_by* (fini par), *is_exactly* (correspond exactement à). Par défaut, la valeur est *like*.

Valeurs de retour

❖ **getRefSchool_byOrgTheResult** (*ArrayOfSchoolStruct* tableau de [SchoolStruct](#))

Tableau d'objet contenant les informations sur les écoles doctorales

1.15. Les projets ANR

Accès au référentiel des projets ANR

getRefCodeANR

Description

Recherche la liste des projets ANR du référentiel

Liste des paramètres

- ❖ **search** (*string*)
Libellé ou code d'un projet recherché

Valeurs de retour

- ❖ **getRefCodeANRResult** (*ArrayOfInts*)
Liste des identifiants des projets ANR

getRefANR_byCode

Description

Retourne les différentes informations d'un projet ANR stockées dans le référentiel

Liste des paramètres

- ❖ **code** (*int*)
Identifiant du projet ANR

Valeurs de retour

- ❖ **getRefANR_byCodeResult** ([ANRStruct](#))
Objet contenant les informations sur un projet ANR

1.16. Les projets européens

Accès au référentiel des projets européens

getRefCodeProjEurope

Description

Recherche la liste des projets européens du référentiel

Liste des paramètres

- ❖ **search** (*string*)
Libellé ou code d'un projet recherché

Valeurs de retour

- ❖ **getRefCodeProjEuropeResult** (*ArrayOfInts*)
Liste des identifiants des projets européens

getRefProjEurope_byCode

Description

Retourne les différentes informations d'un projet européen stockées dans le référentiel

Liste des paramètres

- ❖ **code** (*int*)
Identifiant du projet européen

Valeurs de retour

- ❖ **getRefProjEurope_byCodeResult** (*ProjEuropeStruct*)
Objet contenant les informations sur un projet européen

2. Le dépôt

Chaque dépôt avec texte intégral est vérifié au CCSD (statut **verify** lors du dépôt) avant sa mise en ligne définitive sur l'archive ouverte. HAL propose trois méthodes pour avertir les systèmes locaux après la validation :

- ❖ **Par mail.** HAL envoie un courrier à un email définit entre le CCSD et le système local contenant les identifiants de l'article, le nouveau statut (accept, update, delete) et le motif (si modification ou refus) ;
- ❖ **Par script HTTP méthode POST.** Les paramètres sont l'identifiant (identifiant), la version (version) et le mot de passe de l'article (passwd) dans HAL, le nouveau statut (status : accept, update, delete) et le motif (comment). Un code retour (0 : OK pas de problème, 1 : problème interne et 2 : article inconnu dans le système local) permet à HAL de s'assurer que la mise à jour a pu être effectuée.
- ❖ **Par Web Service.** Le nom de la méthode est *hal_article_status*. Ce WS attendra 5 paramètres en entrée : l'identifiant, la version et le mot de passe de la publication dans HAL, le statut (accept, update, delete) et le motif (si modification ou refus). La chaîne de caractère en retour (local -> 0 : OK pas de problème, 1 : problème interne et 2 : article inconnu) permet à HAL de s'assurer que la mise à jour a pu être effectuée.

L'utilisateur désirent utiliser les WS de dépôt doit dans un premier temps se créer un compte sur un portail de HAL puis demander au ccsd (ccsd-tech@ccsd.cnrs.fr) de lui activer son compte pour l'utilisation des WS. Pour l'environnement de développement, l'utilisateur à utiliser est « test_ws » avec le mot de passe « test ».

2.1. Dépôt d'un nouveau document dans l'archive

upload

Description

Dépôt d'un nouvel article dans l'archive HAL

Liste des paramètres

- ❖ **instance** (*string*)
Instance dans laquelle on souhaite déposer l'article.

La liste des instances disponibles est accessible à partir de la méthode [getRefInstance](#)

❖ **identification** ([UserStruct](#))

Identifiants de connexion sur la plate-forme HAL

❖ **contributor** ([ContributorStruct](#))

Informations personnelles sur le déposant de l'article dans le système local

❖ **metadatas** ([MetasStruct](#))

Métadonnées de l'article déposé. La liste des métadonnées dépend de l'instance de dépôt (liste accessible à partir de la méthode [getRefMetaData](#)) et du type de publication (liste accessible à partir de la méthode [getRefMetaData_forTypePubli](#))

❖ **fullText** ([FullTextStruct](#))

Structure contenant le texte intégral de l'article et l'origine du fichier (fichier auteur, fichier éditeur ; le code est retourné par la méthode [getRefRightFulltext](#))

Valeurs de retour

❖ **uploadResult** ([PendingArticleStruct](#))

Identifiant, version, mot de passe de l'article et état du dépôt dans l'archive (verify, accept, update ou delete)

REMARQUES

- La métadonnée « type de publication » n'est obligatoire que pour le dépôt d'une référence bibliographique (dépôt sans fichier). Si celle-ci n'est pas présente, il s'agit alors d'un document sans références de publication couramment appelé *pre-print* (code = UNDEFINED).
- Si des métadonnées sont multivaluée (domaine scientifique, affiliation des laboratoires, mots clés, etc.) alors chaque valeur est séparée par le caractère « ; ».
- Le format des dates est du type AAAA-MM-JJ, AAAA-MM ou AAAA.

2.2. Correction d'un article

Cette méthode permet de modifier un article pour lequel des modifications ont été demandées par les modérateurs de HAL. Ces modifications ont été transmises au système local selon les 3 méthodes définies plus haut (mail, requête HTTP, WS)

modify

Description

Correction d'un dépôt

Liste des paramètres

- ❖ **article** ([ArticleIdentStruct](#))
Identifiants de l'article sur la plateforme HAL
- ❖ **identification** ([UserStruct](#))
Identifiants de connexion sur la plate-forme HAL
- ❖ **contributor** ([ContributorStruct](#))
Informations personnelles sur le déposant de l'article dans le système local
- ❖ **metadatas** ([MetasStruct](#))
Métadonnées de l'article déposé. La liste des métadonnées dépend de l'instance de dépôt (liste accessible à partir de la méthode [getRefMetaData](#)) et du type de publication (liste accessible à partir de la méthode [getRefMetaData_forTypePubli](#))
- ❖ **fullText** ([FullTextStruct](#))
Structure contenant le texte intégral de l'article et l'origine du fichier (fichier auteur, fichier éditeur ; le code est retourné par la méthode [getRefRightFulltext](#))

Valeurs de retour

- ❖ **modifyResult** ([PendingArticletStruct](#))
Identifiant, version, mot de passe de l'article et état du dépôt dans l'archive (verify, accept, update ou delete)

2.3. Dépôt d'une nouvelle version

Cette méthode permet de déposer une nouvelle version d'un **document avec texte intégral en ligne** dans l'archive HAL. Si les modifications ne portent que sur les métadonnées, il faut utiliser la méthode [update](#).

replace

Description

Dépôt d'une nouvelle version

Liste des paramètres

- ❖ **article** ([ArticleIdentStruct](#))
Identifiants de l'article sur la plateforme HAL
- ❖ **identification** ([UserStruct](#))
Identifiants de connexion sur la plate-forme HAL
- ❖ **contributor** ([ContributorStruct](#))
Informations personnelles sur le déposant de l'article dans le système local
- ❖ **metadatas** ([MetasStruct](#))
Métadonnées de l'article déposé. La liste des métadonnées dépend de l'instance de dépôt (liste accessible à partir de la méthode [getRefMetaData](#)) et du type de publication (liste accessible à partir de la méthode [getRefMetaData_forTypePubli](#))
- ❖ **fullText** ([FullTextStruct](#))
Structure contenant le texte intégral de l'article et l'origine du fichier (fichier auteur, fichier éditeur ; le code est retourné par la méthode [getRefRightFulltext](#))

Valeurs de retour

- ❖ **replaceResult** ([PendingArticleStruct](#))
Identifiant, version, mot de passe de l'article et état du dépôt dans l'archive (verify, accept, update ou delete)

REMARQUES

- La métadonnée « type de publication » est obligatoirement identique à la version précédente sauf si elle était nulle (cas d'un pre-print). Dans ce cas, la nouvelle version peut recevoir des informations de publication.
- La métadonnée « domaine scientifique » ne peut être modifiée lors de la création d'une nouvelle version. Celle-ci n'est donc pas nécessaire dans la liste des métadonnées simples.

- Le code php utilisé pour cette méthode est donc le même que pour un dépôt en remplaçant le paramètre « instance » (string) par le paramètre « article » (ArticleIdentStruct) et en supprimant la métadonnée domaine scientifique.

2.4. Modification des métadonnées

Cette méthode est appelée pour modifier les métadonnées d'un article en ligne sur l'archive HAL.

update

Description

Modification des métadonnées d'un article

Liste des paramètres

- ❖ **article** ([ArticleIdentStruct](#))
Identifiants de l'article sur la plateforme HAL
- ❖ **identification** ([UserStruct](#))
Identifiants de connexion sur la plate-forme HAL
- ❖ **metadatas** ([MetasStruct](#))
Métadonnées de l'article déposé. La liste des métadonnées dépend de l'instance de dépôt (liste accessible à partir de la méthode [getRefMetaData](#)) et du type de publication (liste accessible à partir de la méthode [getRefMetaData_forTypePubli](#))

Valeurs de retour

- ❖ **updateResult** ([ArticleIdentStruct](#))
Identifiant de l'article en ligne sur la plate-forme HAL

REMARQUES

- Les domaines scientifiques ne peuvent pas être modifiés par cette méthode. Il faudra utiliser la méthode [cross](#).

2.5. Ajout / Modification du type de publication

Cette méthode peut être utilisée lorsqu'un article déjà présent dans l'archive HAL est accepté pour publication dans une revue. Dans ce cas, le type de publication change et la méthode `jref` permet d'apporter des informations supplémentaires sur ce nouveau type de publication (nom du journal, éditeur, ...)

jref

Description

Ajout (ou modification) du type de publication du dépôt

Liste des paramètres

- ❖ **article** ([ArticleIdentStruct](#))
Identifiants de l'article sur la plateforme HAL
- ❖ **identification** ([UserStruct](#))
Identifiants de connexion sur la plate-forme HAL
- ❖ **typePubli** (*string*)
Nouveau type de publication (liste accessible à partir de la méthode [getRefTypePubli](#))
- ❖ **metas** (*ArrayOfMetaStruct* : tableau de [MetaStruct](#))
Métadonnées supplémentaires à fournir dépendant de l'instance de dépôt et du type de publication (liste accessible à partir de la méthode [getRefMetaData_forTypePubli](#))

Valeurs de retour

- ❖ **jrefResult** ([ArticleIdentStruct](#))
Identifiant de l'article en ligne sur la plate-forme HAL

2.6. Ajout de domaine scientifique

Un article peut être rattaché à plusieurs domaines scientifiques. Cette méthode va permettre d'ajouter un domaine scientifique à un article déjà présent sur l'archive HAL.

cross

Description

Ajout de domaine scientifique

Liste des paramètres

- ❖ **article** ([ArticleIdentStruct](#))
Identifiants de l'article sur la plateforme HAL
- ❖ **identification** ([UserStruct](#))
Identifiants de connexion sur la plate-forme HAL
- ❖ **domain** (*string*)
Code du domaine à ajouter à l'article. La liste des codes domaines est à récupérer par les méthodes [getRefSubject](#)

Valeurs de retour

- ❖ **crossResult** ([ArticleIdentStruct](#))
Identifiant de l'article en ligne sur la plate-forme HAL

2.7. Ajout de fichier à une notice bibliographique

Cette méthode permet d'ajouter le texte intégral d'un article à une notice bibliographique en ligne sur HAL lorsque celle-ci n'a pas de document attaché.

addFile

Description

Ajout de fichier

Liste des paramètres

- ❖ **article** ([ArticleIdentStruct](#))
Identifiants de l'article sur la plateforme HAL
- ❖ **identification** ([UserStruct](#))
Identifiants de connexion sur la plate-forme HAL
- ❖ **fullText** ([FullTextStruct](#))
Structure contenant le texte intégral de l'article et l'origine du fichier (fichier auteur, fichier éditeur ; le code est retourné par la méthode [getRefRightFulltext](#))

Valeurs de retour

- ❖ **addFileResult** ([PendingArticleStruct](#))
Identifiant, version, mot de passe de l'article et état du dépôt dans l'archive (verify, accept, update ou delete)

REMARQUES

- Attention, cette méthode ne peut être utilisée que si le dépôt est une notice bibliographique (dépôt sans texte intégral).

2.8. Ajout de fichiers annexes à un dépôt avec fichier

Cette méthode permet à un administrateur d'ajouter des fichiers supplémentaires (image ou annexe) à un dépôt avec texte intégral.

addSupplementaryData

Description

Ajout de fichier annexe

Liste des paramètres

- ❖ **article** ([ArticleIdentStruct](#))
Identifiants de l'article sur la plateforme HAL
- ❖ **identification** ([UserStruct](#))
Identifiants de connexion sur la plate-forme HAL
- ❖ **files** ([ArrayOfFileStruct](#))
Structure contenant le(les) fichier(s) annexe(s)

Valeurs de retour

- ❖ **addSupplementaryDataResult** (*booléen*)

REMARQUES

- Attention, cette méthode ne peut être utilisée que par l'administrateur de l'instance du dépôt et que si le dépôt à un fichier (dépôt avec texte intégral).

2.9. Ajout d'un article dans une collection

Cette méthode permet d'ajouter un article dans une collection

addCollection

Description

Tamponnage d'un document

Liste des paramètres

- ❖ **article** ([ArticleIdentStruct](#))
Identifiants de l'article sur la plateforme HAL
- ❖ **identification** ([UserStruct](#))
Identifiants de connexion sur la plate-forme HAL
- ❖ **collection** (*string*)

Identifiant de la collection. La liste des collections peut être récupérée par la méthode [getRefCollection](#)

Valeurs de retour

- ❖ **addCollectionResult** ([ArticleIdentStruct](#))

Identifiant de l'article en ligne sur la plate-forme HAL

2.10. Statut d'un article

Cette méthode permet de récupérer le statut d'un article

status

Description

Récupération du statut d'un document

Liste des paramètres

- ❖ **article** ([ArticleIdentStruct](#))
Identifiants de l'article sur la plateforme HAL
- ❖ **identification** ([UserStruct](#))
Identifiants de connexion sur la plate-forme HAL

Valeurs de retour

- ❖ **statusResult** ([ArticleStatusStruct](#))
Statut de l'article sur la plate-forme HAL : *accept*, *update*, *verify* ou *delete*

2.11. Mise en ligne des fichiers d'un article sous embargo

Méthode permettant de mettre en ligne les fichiers texte intégral d'un article sous embargo dans HAL.

putOnline

Description

Mise en ligne de fichiers

Liste des paramètres

- ❖ **article** ([ArticleIdentStruct](#))
Identifiants de l'article sur la plateforme HAL
- ❖ **identification** ([UserStruct](#))
identifiants de connexion sur la plate-forme HAL

Valeurs de retour

- ❖ **putOnlineResult** (*boolean*)

3. La recherche

3.1. Existence

Recherche de l'existence d'un papier sur HAL

existOnHAL

Description

Recherche dans l'archive HAL d'un papier suivant son titre

Liste des paramètres

- ❖ **title** (*string*)
Le titre du document (le système supprime les chaînes (mots et caractères) de longueur ≤ 2)
- ❖ **status** (*string*)
Critère sur la visibilité de l'article ; peut être visible ou all

Valeurs de retour

- ❖ **existOnHALResult** (*ArrayOfArticleStruct* = tableau de [ArticleStruct](#))
Résultat de la recherche sous la forme d'un tableau d'articles

3.2. Recherche

Consultation/Recherche sur l'archive HAL suivant des critères avancés

search

Description

Recherche dans l'archive HAL

Liste des paramètres

- ❖ **search** (*ArrayOfCritereStruct* = tableau de [CritereStruct](#))
Objet contenant les critères de recherche
- ❖ **typePubli** (*ArrayOfStrings*)
Liste des identifiants (code) des types publications dans lesquelles la recherche va être effectuée.
La liste des types de publications peut être récupérée par la méthode [getRefTypePubli](#)
- ❖ **onlyFullText** (*boolean*)
Recherche uniquement dans les dépôts avec texte intégral si TRUE ; sinon la recherche se fait

sur toute la base.

❖ **orderBy** (*string*)

Critère sur l'ordre de sortie ; peut être date, title ou author

Valeurs de retour

❖ **searchResult** (*ArrayOfArticleStruct* = tableau de [ArticleStruct](#))

Résultat de la recherche sous la forme d'un tableau d'articles

3.3. Récupération des métadonnées d'un article

Cette méthode permet de récupérer les informations d'un article (métadonnées, auteurs) à partir de son identifiant.

getArticleMetadata

Description

Récupération des métadonnées d'un article

Liste des paramètres

- ❖ **identifiant** (*string*)
Identifiant de l'article
- ❖ **version** (*int*)
Version de l'article

Valeurs de retour

- ❖ **getArticleMetadataResult** (*DataSet*)
Structure contenant les informations sur l'article (métadonnées simples, métadonnées auteur)

3.4. Récupération du texte intégral d'un article

Cette méthode permet de récupérer le lien vers le texte intégral (au format PDF ou JPEG) d'un article ou photo présent dans l'archive HAL.

getArticleFile

Description

Récupération du fichier PDF d'un article

Liste des paramètres

- ❖ **identifiant** (*string*)
Identifiant de l'article
- ❖ **version** (*int*)
Version de l'article

Valeurs de retour

- ❖ **getArticleFileResult** (*ArrayOfStrings*)
Tableau contenant le(s) lien(s) vers le(s) fichier(s) PDF ou JPEG correspondant à l'article ou la photo sur la base HAL.

3.5. Recherche de thèses

Recherche de l'existence d'une thèse sur TEL

search_thesis

Description

Recherche de l'existence de thèses dans l'archive TEL

Liste des paramètres

- ❖ **identification** ([UserStruct](#))
Identifiants de connexion privilégiés sur la plate-forme HAL
- ❖ **title** (*string*)
Titre de la thèse
- ❖ **author** (*string*)
Auteur de la thèse
- ❖ **defencedate** (*date*)
Date de soutenance

Valeurs de retour

- ❖ **search_thesisResult** (*ArrayOfArticleStruct* = tableau de [ArticleStruct](#))
Résultat de la recherche sous la forme d'un tableau d'articles

3.6. Récupération des statistiques de consultation d'un article

Cette méthode permet de récupérer les statistiques de consultation d'un article à travers le portail HAL.

getArticleCounter

Description

Récupération du fichier PDF d'un article

Liste des paramètres

- ❖ **identification** ([UserStruct](#))
Identifiants de connexion sur la plate-forme HAL
- ❖ **article** ([ArticleIdentStruct](#))
Identifiants de l'article sur la plateforme HAL

Valeurs de retour

- ❖ **getArticleCounterResult** ([CounterStruct](#))
Statistiques de consultation de l'article : notice/fiche de métadonnées et fichier(s).

4. Les doublons

Service permettant d'informer la base HAL sur l'existence d'éventuel doublon de dépôt

doublon

Liste des paramètres

- ❖ **identification** ([*UserStruct*](#))
Identifiants de connexion sur la plate-forme HAL
- ❖ **identifiants** (*ArrayOfStrings*)
Liste des identifiants des dépôts qui semble être des doublons
- ❖ **identifiant** (*string*)
Parmi les doublons, renseigné ici le dépôt qui vous semble le plus complet et à conserver

Valeurs de retour

- ❖ **doublonResult** (*string*)

ANNEXE : LES STRUCTURES

Identifiant	Structure
ANRStruct	<ul style="list-style-type: none"> ○ code (<i>int</i>) : identifiant HAL du projet ○ reference (<i>string</i>) : référence ANR ○ annee (<i>string</i>) : année du projet ○ acronyme (<i>string</i>) : acronyme ○ titre (<i>string</i>) : titre ○ intitule (<i>string</i>) : intitulé ○ acroappel (<i>string</i>)
ArticleIdentStruct	<ul style="list-style-type: none"> ○ identifiant (<i>string</i>) : identifiant ○ version (<i>int</i>) : version ○ passwd (<i>string</i>) : mot de passe
ArticleStatusStruct	<ul style="list-style-type: none"> ○ status (<i>string</i>) : état de l'article (verify, accept, update ou delete) ○ comment (<i>string</i>) : commentaire permettant d'expliquer la demande de modification ou le refus de l'article
ArticleStruct	<ul style="list-style-type: none"> ○ identifiant (<i>string</i>) : identifiant de l'article dans l'archive hal ○ version (<i>int</i>) : version de l'article ○ url (<i>string</i>) : url stable de l'article ○ title (<i>string</i>) : titre ○ havefile (<i>boolean</i>) : indique si le texte integral de l'article est présent ○ typePubli (<i>string</i>) : type du document
AudienceStruct	<ul style="list-style-type: none"> ○ code (<i>int</i>) : identifiant du type d'audience ○ libelle (<i>string</i>) : libellé

-
- AuthorStruct**
- **labIds** (*ArrayOfInts = tableau d'int*). Chaque entier correspond à la valeur du champ labid d'un objet [LaboratoryStruct](#) (permet d'associer cet auteur à 1 ou plusieurs laboratoires)
 - **lastName** (*string*) : nom
 - **firstName** (*string*) : prénom
 - **otherName** (*string*) : autres prénom, initiales
 - **email** (*string*) : adresse mail de l'auteur
 - **url** (*string*) : url du site perso de l'auteur
 - **organism** (*string*) : établissement de rattachement de l'auteur
 - **researchTeam** (*string*) : équipe de recherche de rattachement de l'auteur
 - **corresponding** (*boolean*) : indique si l'auteur est l'auteur correspondant de la publication

Pour chaque auteur, les données **labIds**, **lastName** et **firstName** sont obligatoires.

- AutLabStruct**
- **authors** (*ArrayOfAuthorStruct = tableau de [AuthorStruct](#)*) : tableau d'auteurs
 - **labs** (*ArrayOfLaboratoryStruct = tableau de [LaboratoryStruct](#)*) : tableau de laboratoires

- CollectionStruct**
- **code** (*string*) : identifiant de la collection
 - **name** (*string*) : intitulé de la collection

- ContributorStruct**
- **lastName** (*string*) : nom
 - **firstName** (*string*) : prénom
 - **email** (*string*) : email

- CounterStruct**
- **metadata** (*int*) : fiche
 - **file** (*int*) : fichier(s)
-

CritereStruct

- **field** (*string*) : nom du champ sur lequel la recherche est faite. La recherche peut être faite sur les champs suivants:
 - *title* : titre de l'article
 - *abstract* : résumé de l'article
 - *author* : auteurs de la publication
 - *lab* : laboratoires associés aux auteurs de l'article
 - *researchteam* : équipe de recherche des auteurs
 - *collaboration* : collaboration
 - *collection* : identifiant du tampon de l'article
 - *year* : année de publication ou année d'écriture ou année de dépôt
- **searchType** (*string*) : type de recherche. Ce paramètre peut prendre les valeurs
 - *like* : contient (valeur par défaut)
 - *start_by* : commence par
 - *end_by* : fini par
 - *is_exactly* : correspond exactement à
 - *gt* : supérieur à
 - *lt* : inférieur à
 - *ge* : supérieur ou égal à
 - *le* : inférieur ou égal à
- **value** (*string*) : texte recherché
- **join** (*string*) : Utilisé pour joindre plusieurs critères de recherche. Ce paramètre peut prendre les valeurs :
 - *and* : et (valeur par défaut)
 - *or* : ou
 - *and_not* : sauf

DateVisibleStruct

- **code** (*string*) : durée d'embargo
- **libelle** (*string*) : libellé

FileStruct

- **name** (*string*) : nom du fichier
- **md5** (*string*) : signature md5 du contenu du fichier non encodé en base 64
- **format** (*string*) : code
- **mainWebFile** (*string*) : fichier principal lors d'un dépôt au format HTML ou XML
- **content** (*string*) : contenu du fichier encode en base 64 ou url du fichier
- **local** (*boolean*) : si local= true alors content correspond au contenu du fichier encode en base 64, sinon content correspond à l'url du fichier
- **mainWebFile** (*string*) : permet de renseigner le fichier principal lors d'un transfert de papier au format HTML, XML ou IMG.

Si le paramètre **md5** n'est pas vide, celui-ci permet de vérifier que le contenu du fichier est correct après transfert sur HAL. Le dépôt d'archive zip, tar (par format de fichiers) est bien sûr possible.

FormatStruct

- **code** (*string*) : identifiant du format
- **libelle** (*string*) : libellé

FullTextStruct

- **right** (*int*) : droits sur les fichiers déposes
- **files** (*ArrayOfFileStruct* = tableau de [FileStruct](#)) : tableau des différents fichiers déposés

- LaboratoryStruct**
- **labId** (*int*) : identifiant interne à la structure [AutlabStruct](#) (permet d'effectuer les associations auteurs / laboratoires)
 - **knownLabId** (*int*) : identifiant HAL du laboratoire
 - **shortName** (*string*) : sigle
 - **name** (*string*) : nom
 - **address** (*string*) : adresse postale
 - **url** (*string*) : url du site web
 - **pays** (*string*) : code ISO 3166 alpha-2
 - **affiliation** (*ArrayOfStrings* = *tableau de string*) : affiliations du laboratoire

Lorsque l'on renseigne un laboratoire « connu » de HAL, **knownLabId** est le seul champ obligatoire. Sinon, **name**, **pays** et **au moins une affiliation** sont obligatoires pour renseigner un nouveau laboratoire. La donnée **LabId** est obligatoire puisque servant à associer le laboratoire à un ou plusieurs auteurs.

- LabStruct**
- **id** (*int*) : Identifiant unique du laboratoire
 - **name** (*string*) : Nom du laboratoire
 - **shortName** (*string*) : Sigle du laboratoire
 - **address** (*string*) : Coordonnées postales du labo
 - **url** (*string*) : adresse du site web du labo
 - **pays** (*string*) : identifiant du pays (code ISO 3166-1)
 - **affiliation** (*ArrayOfStrings*) : affiliations du laboratoire
 - **status** (*string*) : état du laboratoire [open, close ou tmp]

- MetaDataStruct**
- **name** (*string*) : identifiant
 - **libelle** (*string*) : libelle
 - **comment** (*string*) : descriptif
 - **required** (*boolean*) : indique si la métadonnée est obligatoire ou pas
 - **domain** (*ArrayOfStrings*) : codes des domaines pour lesquels la métadonnée est à renseigner (ALL correspond à tous les domaines scientifiques)

- DatasStruct**
- **metaSimple** (*ArrayOfMetaStruct* = tableau de [MetaStruct](#)) : tableau contenant les métadonnées de l'article
 - **metaAutLab** ([AutLabStruct](#)) : liste des auteurs d'un article

- MetaStruct**
- **metaName** (*string*) : nom de la métadonnée (titre, résumé, ...)
 - **metaCode** (*string*) : code si la méta en possède (ex langue, domaine, ...)
 - **metaValue** (*string*) : valeur de la métadonnée

- OrgTheStruct**
- **name** (*string*) : nom de l'organisme
 - **academy** (*string*) : académie
 - **url** (*string*) : adresse du site Web de l'organisme
 - **pays** (*string*) : code ISO 3166-1 du pays de l'organisme

- PendingArticleStruct**
- **identifiant** (*string*) : identifiant
 - **version** (*int*) : version
 - **passwd** (*string*) : mot de passe
 - **status** (*string*) : état de l'article :
 - verify : en vérification
 - accept : accepté
 - update : demande de modification
 - delete : supprimé

- ProjEuropeStruct**
- **code** (*int*) : identifiant HAL du projet
 - **numero** (*string*) : numéro du projet
 - **acronyme** (*string*) : acronyme
 - **titre** (*string*) : titre
 - **callid** (*string*)
 - **fundedby** (*string*)
 - **sdate** (*date*) : date début projet
 - **edate** (*date*) : date fin projet

ReviewStruct	<ul style="list-style-type: none">○ name (<i>string</i>) : nom complet de la revue○ shortName (<i>string</i>) : nom abrégé de la revue (s'il existe et est connu du référentiel)○ ISSN (<i>string</i>) : ISSN de la revue (si connu du référentiel)○ editor (<i>string</i>) : éditeur de la revue (si connu du référentiel)○ EISSN (<i>string</i>) : ISSN de la revue version électronique (s'il existe et est connu du référentiel)
---------------------	---

RightFulltextStruct	<ul style="list-style-type: none">○ code (<i>string</i>) : droit○ libelle (<i>string</i>) : libellé
----------------------------	--

SchoolStruct	<ul style="list-style-type: none">○ name (<i>string</i>) : nom de l'école○ orgThe (<i>string</i>) : organisme de thèse
---------------------	---

SubjectStruct	<ul style="list-style-type: none">○ code (<i>string</i>) : identifiant du domaine○ codeParent(<i>string</i>) : code 'père' du domaine○ deposable (<i>boolean</i>) : prend les valeurs 'TRUE' ou 'FALSE' suivant s'il est permis de déposer dans ce domaine ou pas.○ haveNext (<i>boolean</i>) : prend les valeurs 'TRUE' ou 'FALSE'○ libelle (<i>string</i>) : libellé d'un domaine
----------------------	--

TypePubliStruct	<ul style="list-style-type: none">○ code (<i>string</i>): identifiant○ libelle (<i>string</i>) : libelle○ type (<i>string</i>) : all, file ou notice.
------------------------	--

UserStruct	<ul style="list-style-type: none">○ uid (<i>string</i>): login de connexion○ passwd (<i>string</i>) : mot de passe
-------------------	---
