

Fonctions générales

Les fonctions générales regroupent des fonctions qui permettent :

- L'initialisation des transactions via l'API
- Le transfert de données

GedInitConnection

GedInitConnection initialise l'interconnexion et prépare une transaction.

Cette fonction doit obligatoirement être appelée au début de chaque transaction.

```
int GedInitConnection(  
    char *GedAdresse,  
    char *GedPort,  
    int ModeTrace,  
    char *LogFile  
);
```

Paramètres

GedAdresse : DNS ou IP pour atteindre l'automate Multigest.

GedPort : Port d'écoute (listen) de l'automate Multigest.

ModeTrace : Activation du mode « DEBUG » : 0 / 1 (0 par défaut).

Logfile : Fichier d'écriture des traces

Codes retour

Cette fonction retourne toujours 1

Le mode « DEBUG » écrit des traces dans le fichier passé dans le paramètre « Logfile ». Si ce paramètre est omis, le fichier par défaut sera « c:\AxbdDll.log ».

GedDebug

GedDebug permet d'activer ou de désactiver à la volée les traces des appels.

```
void GedDebug(  
    int state  
) ;
```

Paramètres

state : État d'activation des traces 0 : inactives / 1 : actives

Codes retour

Aucun



Information

Le mode « DEBUG » écrit des traces dans le fichier php.log définit dans le paramétrage de PHP.

GedAddChampRecherche

GedAddChampRecherche permet de passer à l'API les champs de recherche permettant d'identifier **un** dossier GED. Afin d'identifier ces dossiers, il est nécessaire de définir une clé de recherche et les valeurs à rechercher.



Information

Si la clé de recherche est multiple, cette fonction doit être appelée pour chaque champ de recherche.

```
void GedAddChampRecherche (  
    char *FieldName,  
    char *FieldValue  
);
```

Paramètres

FieldName : Nom physique du champ de recherche

FieldValue : Valeur de recherche

GedAddMultiChampRecherche

GedAddMultiChampRecherche permet de passer à l'API les champs de recherche permettant d'identifier un dossier GED. Afin d'identifier ces dossiers, il est nécessaire de définir une clé de recherche et les valeurs à rechercher.

```
void GedAddMultiChampRecherche(  
    char *FieldNames,  
    char *FieldValues  
);
```

Paramètres

FieldName : Nom des champs (séparateur |)

FieldValue : Valeurs des champs (séparateur |)

GedAddChampRequete

GedAddChampRequete permet de passer à l'API les champs d'une armoire Multigest (Nom physique du champ et sa valeur) pour la création ou la modification des fiches d'index des dossiers GED (Création de dossiers [INSERT] ou modification de dossier [UPDATE]).

```
void GedAddChampRequete(  
    char *FieldName,  
    char *FieldValue  
);
```

Paramètres

FieldName : Nom physique du champ de recherche

FieldValue : Valeur de recherche

GedAddMultiChampRequete

GedAddMultiChampRequete permet de passer à l'API les champs d'une armoire Multigest (Nom physique du champ et sa valeur) pour la création ou la modification des fiches d'index des dossiers GED (Création de dossiers [INSERT] ou modification de dossier [UPDATE]).

```
void GedAddMultiChampRequete(  
    char *FieldNames,  
    char *FieldValues  
);
```

Paramètres

FieldName : Nom des champs (séparateur |)

FieldValue : Valeurs des champs (séparateur |)

GedChampReset

GedChampReset permet de remettre à zéro les critères passés par GedAddChampRecherche / GedAddChampRequete / GedAddMultiChampRecherche / GedAddMultiChampRequete.

```
void GedChampReset();
```

GedSetModeUid

GedSetModeUid permet l'activation du mode UID.

Ce mode permet l'utilisation de l'Identifiant Unique de Document dans les paramètres des méthodes en lieu et place de l'identifiant système de document -ou plus précisément en lieu et place du binôme « identifiant système du conteneur (armoire, bannette)/ identifiant système de document »-.

Ce mode impacte aussi les valeurs de retour puisque l'identifiant système de document est remplacé par l'UID lorsque le mode est activé.

Notion sur l'UID :

Identifiant Unique de Document attribué au document à son entrée en GED indifféremment de son point d'entrée (dans une armoire, une bannette, une bannette mail) et qu'il conservera tout au long de son cycle de vie. Ceci contrairement à son identifiant système qui est modifié lors de certaines opérations : déplacement du document d'une armoire à une autre, indexation du document depuis une bannette vers une armoire, etc...

L'UID utilisé seul est suffisant pour déterminer de quel document il s'agit, alors que l'identifiant système de document nécessite la plupart du temps une utilisation par binôme :

- Si le document se trouve dans une armoire :

Identifiant système de l'armoire / identifiant système de document

- Si le document se trouve dans une bannette :

Identifiant système de la bannette / identifiant système de document

Ce mode est désactivé par défaut.

```
void GedSetModeUid (  
    int *state  
);
```

Paramètres

state : 0 désactivation du mode. 1 activation du mode