

WebServices EPN

Documentation technique

Rédacteur(s)

Guillaume MESSONNIER - Decalog

Edité le : 20/01/2017

TABLE DES MATIERES

1. Principe général	2
2. Webservice n°1 : Informations générales	3
2.1. Requête.....	3
2.2. Réponse	3
2.3. Exemple PHP	4
2.4. Exemple Java.....	4
3. Webservice n°2 : Authentification d'un abonné	5
3.1. Requête.....	5
3.2. Réponse	5
3.3. Exemple PHP	6
3.4. Exemple Java.....	7
4. Webservice n°3 : Liste des abonnés modifiés	8
4.1. Requête.....	8
4.2. Réponse	8
4.3. Exemple PHP	9
4.4. Exemple Java.....	10
5. Annexe n°1 : Script de test PHP	11

5.1.	Pré-requis.....	11
5.2.	Fonctions utilitaires.....	11
5.3.	WebService n°1 : Informations système	11
5.4.	WebService n°2 : Authentification d'un abonné	12
5.5.	WebService n°3 : Liste des abonnés modifiés.....	12

1. Principe général

Decalog SIGB sont des plateformes de services de gestion de bibliothèque en ligne. Elles fournissent un ensemble de WebServices permettant d'interagir avec elles : il s'agit de WebServices REST (Representational State Transfer) interrogeable au travers du protocole HTTP et du format de données JSON.

L'utilisation de cette architecture ouverte alliée à des protocoles et des formats de données standardisés font de Decalog SIGB des plateformes de services interopérables. Ainsi, les WebServices fournis peuvent être interrogés à partir de nombreux langages (PHP, Java, Perl, Python, C#, C/C++, etc.)

Pour interroger ces WebServices, **le client doit au préalable avoir obtenu un point d'accès et une clé d'authentification auprès de la société Decalog**. Pour cela, il suffit de contacter le support logiciel Decalog ou le responsable d'exploitation dont les coordonnées sont en page de garde.

Attention, dans la suite de ce document, nous utiliserons systématiquement l'URL <https://sigb01.decalog.net> comme point d'accès. Néanmoins, l'URL réelle que vous devrez utiliser vous sera communiquée en même temps de la clé d'authentification.

2. Webservice n°1 : Informations générales

2.1. Requête

URL du Webservice	/ws.php?_action_=epm&path=/system
Entête HTTP	Clé d'authentification au format « apikey:clé_api »
Format de la réponse	JSON

Pour avoir l'autorisation d'interroger ce Webservice, il est nécessaire d'envoyer la clé d'authentification qui vous a été transmise dans les entêtes (« headers ») HTTP sous le format « **apikey:clé_api_fourni** ».

2.2. Réponse

La réponse du Webservice est au format JSON :

success	1 en cas de succès, 0 si l'appel du Webservice a échoué.
res	Le résultat du Webservice détaillé ci-dessous

L'objet a la structure suivante :

id	Identifiant unique de la base du client										
name	Nom de la base du client										
type	• MONO_SITE : Client avec une seule bibliothèque • MULTI_SITES : Client avec plusieurs bibliothèques										
subs	Tableau des cotisations déclarées dans la base : <table> <tr> <td>id</td><td>Identifiant unique de la cotisation</td></tr> <tr> <td>name</td><td>Libellé de la cotisation</td></tr> <tr> <td>restrictedLibs</td><td> Tableau des sites sur lesquels est utilisable la cotisation : <table> <tr> <td>id</td><td>Identifiant unique du site</td></tr> <tr> <td>name</td><td>Libellé du site</td></tr> </table> </td></tr> </table>	id	Identifiant unique de la cotisation	name	Libellé de la cotisation	restrictedLibs	Tableau des sites sur lesquels est utilisable la cotisation : <table> <tr> <td>id</td><td>Identifiant unique du site</td></tr> <tr> <td>name</td><td>Libellé du site</td></tr> </table>	id	Identifiant unique du site	name	Libellé du site
id	Identifiant unique de la cotisation										
name	Libellé de la cotisation										
restrictedLibs	Tableau des sites sur lesquels est utilisable la cotisation : <table> <tr> <td>id</td><td>Identifiant unique du site</td></tr> <tr> <td>name</td><td>Libellé du site</td></tr> </table>	id	Identifiant unique du site	name	Libellé du site						
id	Identifiant unique du site										
name	Libellé du site										

2.3. Exemple PHP

La trace exécution suivante a été obtenue en utilisant le script PHP disponible en annexe 1.

https://sigb01.decalog.net/ws.php?action_=epm&path=/system

```
stdClass Object
(
    [success] => 1
    [res] => stdClass Object
        (
            [id] => 3392971985437509977
            [name] => Valence
            [type] => MULTI_SITES
            [subs] => Array
                (
                    [0] => stdClass Object
                        (
                            [restrictedLibs] => Array
                                (
                                    [0] => stdClass Object
                                        (
                                            [id] => 3879887951953438255
                                            [name] => Site1
                                        )
                                    [1] => stdClass Object
                                        (
                                            [id] => 203356197907233335
                                            [name] => Site2
                                        )
                                )
                            [name] => Cotisation
                            [id] => 8690059263494048095
                        )
                )
        )
)
```

2.4. Exemple Java

```
{
  "success":true,
  "res": {
    "id":"3069072133511143166",
    "name":"Adultes"
    "subs":[{
      "restrictedLibs":[{
        "id":"730562953594818526"
        "name":"Biblioth\u00e8que Jean Marc Reiser",
      }],
    }],
  }
}
```

3. Webservice n°2 : Authentification d'un abonné

3.1. Requête

URL du Webservice	/ws.php?_action_=epm&path=/user
Entêtes HTTP	• Clé d'authentification au format « apikey:clé_api » • Identifiants de l'abonné au format « Authorization: Basic login:passwd » où « <i>login:passwd</i> » est encodé en base 64
Format de la réponse	JSON

Pour avoir l'autorisation d'interroger ce Webservice, il est nécessaire d'envoyer dans les entêtes (« headers ») HTTP la clé d'authentification qui vous a été transmise sous le format « **apikey:clé_api_fourni** ».

Les identifiants de l'abonné que vous essayez d'authentifier doivent eux aussi être transmis par le biais des entêtes http au format « **Authorization: Basic login:passwd** » où la chaîne de caractères « *login:passwd* » est encodé en **base 64**.

Exemple : « Authorization: Basic QWxhZGRpbjpvcGVuIHNlc2FtZQ== »

3.2. Réponse

La réponse du Webservice est au format JSON :

success	• 1 si l'authentification réussie (identifiants de connexion corrects) • 0 si l'authentification échoue
res	Le résultat du Webservice détaillé ci-dessous

L'objet **res** a la structure suivante :

id	Identifiant unique de la base du client
code	Code-barres de l'abonné
lastname	Patronyme de l'abonné
secondaryname	Prénom de l'abonné
birthDate	Date de naissance de l'abonné au format ISO8601
sex	Sexe de l'abonné (M pour masculin, F pour féminin)
email	Courriel de l'abonné

subscriptionDate	Date de première inscription au formation ISO8601								
deleted	Abonné supprimé ou non (booléen)								
login	Identifiant de connexion de l'abonné								
subs	Tableau des cotisations de l'abonné : <table> <tr> <td>subscriptionId</td><td>Identifiant de la cotisation</td></tr> <tr> <td>subscriptionName</td><td>Libellé de la cotisation</td></tr> <tr> <td>startDate</td><td>Date de début de validité (ISO8601)</td></tr> <tr> <td>endDate</td><td>Date de fin de validité (ISO8601)</td></tr> </table>	subscriptionId	Identifiant de la cotisation	subscriptionName	Libellé de la cotisation	startDate	Date de début de validité (ISO8601)	endDate	Date de fin de validité (ISO8601)
subscriptionId	Identifiant de la cotisation								
subscriptionName	Libellé de la cotisation								
startDate	Date de début de validité (ISO8601)								
endDate	Date de fin de validité (ISO8601)								

3.3. Exemple PHP

La trace exécution suivante a été obtenue en utilisant le script PHP disponible en annexe 1.

https://sigb01.decalog.net/ws.php?_action_=epm&path=/user

```
stdClass Object
(
    [success] => 1
    [res] => stdClass Object
    (
        [id] => 7413781156004930397
        [code] => A001
        [lastname] => Ochon
        [secondaryname] => Paul
        [birthDate] => 1960-06-12T00:00:00
        [sex] => F
        [email] =>
        [subscriptionDate] => 2009-10-12T00:00:00
        [deleted] =>
        [login] => p.ochon
        [subs] => Array
        (
            [0] => stdClass Object
            (
                [subscriptionId] => 8690059263494048095
                [subscriptionName] => Cotisation Adultes
                [startDate] => 2013-07-23T00:00:00
                [endDate] => 2014-07-23T00:00:00
            )
        )
    )
)
```

3.4. Exemple Java

```
{
  "success":true,
  "res":{
    "id":"4586781712495406414",
    "secondaryname":"Sophie",
    "lastname":"Stiquet",
    "code":"847",
    "birthDate":null,
    "subs":[{
      "subscriptionId":"3069072133511143166",
      "startDate":"2013-06-26T00:00:00",
      "endDate":"2014-06-26T00:00:00"
    }]
  }
}
```


4. Webservice n°3 : Liste des abonnés modifiés

4.1. Requête

URL du Webservice	/ws.php?_action_=epm&path=/modifiedusers/{date}
Entête HTTP	Clé d'authentification au format « apikey:clé_api »
Format de la réponse	JSON

Pour avoir l'autorisation d'interroger ce Webservice, il est nécessaire d'envoyer dans les entêtes (« headers ») HTTP la clé d'authentification qui vous a été transmise sous le format « **apikey:clé_api_fourni** ».

4.2. Réponse

La réponse du Webservice est au format JSON :

success	1 en cas de succès, 0 si l'appel du Webservice a échoué.
res	Le résultat du Webservice détaillé ci-dessous

L'objet **res** a la structure suivante :

id	Identifiant unique de la base du client
code	Code-barres de l'abonné
lastname	Patronyme de l'abonné
secondaryname	Prénom de l'abonné
birthDate	Date de naissance de l'abonné au format ISO8601
sex	Sexe de l'abonné (M pour masculin, F pour féminin)
email	Courriel de l'abonné
subscriptionDate	Date de première inscription au format ISO8601
deleted	Abonné supprimé ou non (booléen)
login	Identifiant de connexion de l'abonné
subs	Tableau des cotisations de l'abonné :

	subscriptionId	Identifiant de la cotisation
	subscriptionName	Libellé de la cotisation
	startDate	Date de début de validité (ISO8601)
	endDate	Date de fin de validité (ISO8601)

4.3. Exemple PHP

La trace exécution suivante a été obtenue en utilisant le script PHP disponible en annexe 1, elle liste les fiches abonnés modifiées depuis le 15 juillet 2013 à 15:25:35.

https://sigb01.decalog.net/ws.php?_action_=epm&path=/modifiedusers/20130731152535

```
stdClass Object
(
    [success] => 1
    [res] => Array
        (
            [0] => stdClass Object
                (
                    [id] => 7413781156004930397
                    [code] => A001
                    [lastname] => Ochon
                    [secondaryname] => Paul
                    [birthDate] => 1960-06-12T00:00:00
                    [sex] => F
                    [email] =>
                    [subscriptionDate] => 2009-10-12T00:00:00
                    [deleted] =>
                    [login] => p.ochon
                    [subs] => Array
                        (
                            [0] => stdClass Object
                                (
                                    [subscriptionId] => 8690059263494048095
                                    [subscriptionName] => Cotisation Adultes
                                    [startDate] => 2013-07-23T00:00:00
                                    [endDate] => 2014-07-23T00:00:00
                                )
                            )
                        )
                )
        )
    )
)
```

4.4. Exemple Java

```
{
  "success":true,
  "res":{
    "id":"4586781712495406414",
    "secondaryname":"Sophie",
    "lastname":"Stiquet",
    "code":"847",
    "birthDate":null,
    "subs":[{
      "subscriptionId":"3069072133511143166",
      "startDate":"2013-06-26T00:00:00",
      "endDate":"2014-06-26T00:00:00"
    }]
  }
}
```

5. Annexe n°1 : Script de test PHP

5.1. Pré-requis

Ce script utilise la librairie **cURL** pour effectuer les appels aux WebServices. Ainsi, pour l'utiliser, vous devez au préalable vérifier que l'extension cURL est bien installée sur votre version de PHP. Curl est une bibliothèque qui permet de faire des appels https avec des headers.

5.2. Fonctions utilitaires

```
<?php

//prepare curl for a request (i.e content only, no ssl verification)
function curl_eppk_prepare($url, $apikey, $login = null, $password = null) {
    $session = curl_init($url);
    curl_setopt($session, CURLOPT_HEADER, false);
    curl_setopt($session, CURLOPT_RETURNTRANSFER, true);
    curl_setopt($session, CURLOPT_SSL_VERIFYHOST, false);
    curl_setopt($session, CURLOPT_SSL_VERIFYPEER, false);
    if ($login && $password) {
        curl_setopt($session, CURLOPT_USERPWD, sprintf("%s:%s", $login, $password));
    }
    curl_setopt($session, CURLOPT_HTTPHEADER, array(sprintf('apikey:%s', $apikey)));
    return $session;
}

function do_get($url, $apikey, $login, $password) {
    $session = curl_eppk_prepare($url, $apikey, $login, $password);
    $response = curl_exec($session);
    curl_close($session);
    return $response;
}

function buildUrl($base_url, $path) {
    return sprintf($base_url, urlencode($path));
}
```

5.3. Webservice n°1 : Informations système

```
/*
 * Configuration
 */
$base_url = 'https://sigb01.decalog.net/ws.php?action=eppm&path=%s';
$apikey = 'my_key';

/*
 * Main program
 */
$systemUrl = buildUrl($base_url, '/system');
$system = json_decode(do_get($systemUrl, $apikey));
echo "$systemUrl\n\n";
```

```
print_r($system);
```

5.4. Webservice n°2 : Authentification d'un abonné

```
/*
 * Configuration
 */
$base_url = 'https://sigb01.decalog.net/ws.php?_action_=epm&path=%s';
$apikey = 'my_key';
$login = 'my_login';
$password = 'my_password';

/*
 * Main program
 */
$userUrl = buildUrl($base_url, '/user');
$user = json_decode(do_get($userUrl, $apikey, $login, $password));
echo "$userUrl\n\n";
print_r($user);
```

5.5. Webservice n°3 : Liste des abonnés modifiés

```
/*
 * Configuration
 */
$base_url = 'https://sigb01.decalog.net/ws.php?_action_=epm&path=%s';
$apikey = 'my_key';
$date = '20130731152535'; // Format : YYYYMMDDHHMMSS (Année-Mois-Jour-Heures-Minutes-Secondes)

/*
 * Main program
 */
$userModUrl = buildUrl($base_url, '/modifiedusers/'.$date);
$userMod = json_decode(do_get($userModUrl, $apikey));
echo "$userModUrl\n\n";
print_r($userMod);
```