



API-Format des échanges

	Version	V0.3
Référence	S-money-API-Format des échanges	
Objet	Spécifications	

Contenu

1. FORMAT DES REQUETES ET REPONSES DE L'API.....	2
1.1.1 Exemple d'appel en JSON :.....	2
1.1.2 Exemple de retour en JSON :	2
1.1.3 Codes HTTP retournés.....	2
1.1.4 Format des erreurs.....	2
2. VERSIONING	3
3. PAGINATION.....	3



1. FORMAT DES REQUETES ET REPONSES DE L'API

Pour communiquer avec l'API, il est conseillé d'utiliser le format JSON en passant la valeur json aux headers HTTP Accept et Content-Type.

Il reste toutefois possible de communiquer au format XML en passant la valeur xml.

1.1.1 Exemple d'un appel GET en JSON :

GET [/api/\[domaine_partenaire\]/subaccounts](#) HTTP/1.1

Accept: application/vnd.s-money.v1+json

Authorization: Bearer ACCESS_TOKEN

1.1.2 Exemple d'un appel POST en JSON :

POST [/api/\[domaine_partenaire\]/subaccounts](#) HTTP/1.1

Accept: application/vnd.s-money.v1+json

Content-Type: application/vnd.s-money.v1+json

Authorization: Bearer ACCESS_TOKEN

```
{"appaccountid":123, "displayname":"Exemple de sous-compte"}
```

1.1.3 Exemple de retour en JSON :

HTTP/1.1 201 CREATED

Content-Type: application/vnd.s-money.v1+json

Location: /api/[domaine_partenaire]/subaccounts/123

```
{"appaccountid":"123","displayname":"Exemple de sous-compte"}
```

1.1.4 Codes HTTP retournés

En cas de succès :

- 200 OK
- 201 CREATED
- 204 NO CONTENT

En cas d'erreur :

- 400 BAD REQUEST
- 401 UNAUTHORIZED
- 403 FORBIDDEN
- 404 NOT FOUND
- 500 INTERNAL SERVER ERROR

1.1.5 Format des erreurs

Dans les cas d'erreurs, les détails sont retournés sous le format suivant :

```
{  
  "Code":513,
```



```
"ErrorMessage": "Sous-compte introuvable.",  
"Title": "L'opération ne peut pas aboutir",  
"Priority": 2  
}
```

Les différentes priorités des erreurs sont définies comme suit :

- 0 : reset
- 1 : fatal
- 2 : bloquant
- 3 : non bloquant

2. VERSIONING

Le header Accept est obligatoire pour chaque requête à l'API. Ce header permet de préciser la version de la ressource que le client attend en réponse de la requête :

Accept: application/vnd.s-money.v1+json

ou

Accept: application/vnd.s-money.v1+xml

De même, pour les requêtes POST ou PUT, il faut préciser la version de la ressource fournie dans la requête par le client dans le header Content-Type :

Content-Type: application/vnd.s-money.v1+json

ou

Content-Type: application/vnd.s-money.v1+xml

Le serveur répondra en fonction de la version passée dans ce header. La version retournée par le serveur est précisée dans le header Content-Type :

Content-Type: application/vnd.s-money.v1+json

ou

Content-Type: application/vnd.s-money.v1+xml

3. PAGINATION

Les requêtes qui retournent plusieurs résultats sont paginées par défaut à 50 résultats. Pour naviguer vers une page spécifique, il faut spécifier le paramètre page dans les paramètres de la requête. Il est également possible de définir le nombre de résultats par page en incluant le paramètre perpage.

```
GET /api/[domaine_partenaire]/historyitems?page=2&perpage=30
```

La numérotation des pages commence à 1. Si le numéro de page n'est pas spécifié, la requête retourne par défaut la première page.

Les informations de pagination sont incluses dans les headers HTTP de la réponse. Le header Link (RFC 5988) retourne les url des pages précédentes et suivantes :

```
Link: </api/[domaine_partenaire]/historyitems?page=2&perpage=30>; rel="next",  
</api/[domaine_partenaire]/historyitems?page=5&perpage=30>; rel="last"
```

Les valeurs de rel possibles sont :

Ce document est la propriété de BPCE. Ne peut être communiqué à des tiers sans autorisation écrite.



next : contient l'url de la page suivante (non renseigné si elle n'existe pas).

last : contient l'url de la dernière page (non renseigné si déjà sur la dernière page).

prev : contient l'url de la page précédente (non renseigné si elle n'existe pas).

first : contient l'url de la première page (non renseigné si déjà sur la première page).

Le header Total-count contient quant à lui le nombre total d'éléments.