

Le design d'API REST, un débat sans fin ?

Speaker : Guillaume Laforge (Restlet)

Format : Conférence

Date : 22 avril 2016

Restlet propose plusieurs produits : Studio, APISpark, Framework, DHC



Roy Fielding a défini Representational State Transfer :

- Identification des ressources : URIs
- Représentation : JSON, XML
- Self-descriptive message : http, GET, cacheability
- HAETIOAS

Les verbes permettent de travailler à la fois sur une ressource ou une liste de ressources.
Pas de POST pour un item mais sur la liste.

Nommage

Pour les ressources, **privilégiez l'usage des noms plutôt que des verbes** (différents qu'en SOAP).

Les verbes peuvent être utilisés pour les actions ou les calculs : /login, /convertTemperature

Préférez l'usage du pluriel. Ex : /tickets/234

Permet de s'affranchir des formes singulières / pluriel : /person vs /people

Je veux le 234^{ième} élément de la collection tickets

Forme de nommage : UpperCamelCase, lowerCamelCase, snake_case, dashed-nake-case

Question de goût. Choisir une norme et rester cohérent

Gestion des relations

Notion de sous-ressources : /tickets/123/messages/4

Lorsqu'une ressource est indépendante, avoir sa propre url.

Ex : /usergroups/24/users/12 => /users/12

Paramètres d'API

Pas de règles en général. Celles utilisées chez Restlet :

- Path : Required, resource identifier
- Query : optional, query collections (ex : pagination)
- Body : resource specific logic
- Header : global, platform-wide (ex : API Keys)

HTTP Status Code

- 4xx : le client de l'API a fait une erreur
- 5xx : erreur côté serveur
- Lors de la pagination, on pourrait utiliser 206 partial content + Link : dans les headers pour les URL prev/net
- Lors de la création d'une nouvelle ressource, renvoyer un 201 created avec le header Location
- Lors d'un traitement asynchrone, on peut renvoyer 202 Accepted (no payload returned)
- Après un DELETE, retourner 204 No content

Anti-pattern : Facebook renvoie toujours 200 OK même en cas d'erreur

API Navigation

Démo avec DHC et l'API Starwars

Caching

Etag avec les hashtag

Ou Not-Modified-Since

Pagination

En paramètre de requêtes HTTP :

Page number : ?page=23,

Avec des curseurs : ?cursor=34ea3fd6 (insertion-friendly) pour ne pas rater des éléments

Pagination par attribut

Rare mais élégant : Accept range header not just for bytes

GET /users

Accept-Ranges : users

Content-Range users 0-9/200

Collections

Préférer les unwrapped collection à la place des données (objet intermédiaire data), cela permet de rajouter des méta-données

Erreurs

Standardisation en cours du contenu de la réponse d'erreur (cf. photo ci-dessus).

Si une API retourne un code d'erreur 4xxxx non supporté par le client, ce dernier doit le traiter comme une erreur générique 400.

Filtering

Les besoins mobiles et web peuvent différer.
On peut filtrer les données qu'on souhaite récupérer.

Avec les query parameters :

?fields=firstname,lastname,age

?exclude=biography,style

?style=compact

Avec le header prefer

Prefer : return=minimal

Utilisation du caractère point . pour récupérer le sous-champs :

?fields=address.zip

Tri

- SQL-style : ?sort=title+DESC
- Sort + asc/desc COMBO

Recherche / Filter / sort

API permettant au client de récupérer ce qu'ils souhaitent :

Facebook's GraphQL

Netflix's Falcor

Versionning d'API

- Le plus fréquent dans l'URL : v1
- Custom Header : X-API-Version : 2
- Sur GitHub : MIMEType avec la version

HyperMedia

Richardson a défini un modèle de maturité.

Le dernier niveau, le level 3 concerne le Hypermedia Controls.

Avantages :

- Clients plus génériques
- Plus besoin de versionner systématiquement l'API, par exemple un changement d'URL d'une ressource

Inconvénients :

- Réponse plus lourde (terminaux mobiles et mauvaise connexion)
- Les clients doivent connaître à quoi correspond le qualificatif du lien retourné

Plein d'approches : HAL, JSON-LD ...

Avec HAL, on ajoute un attribut `_links`

Identifiants

Utiliser les UUID à la place des identifiants séquentiels

Questions / Réponses

- Verbe PATCH : mise à jour partielle (ex : un seul champ). Regarder JSON-PATCH pour faire des diff entre les payloads