JSON brièvement

Définition

- JSON : **J**ava**S**cript **O**bject **N**otation => le langage JavaScript, en effet la syntaxe de JSON reprend la syntaxe de création d'objets en JavaScript.
- Créé par Douglas Crockford, 2005
- JSON désigne un format de données (textuel léger) permettant de stocker de l'information de manière organisée => Même objectif que XML, mais plus simple
- Pas un langage de markups
- Il est souvent utilisé pour décrire des structures de type objet.
- A l'origine pour JavaScript => Indépendant du langage de programmation : JavaScript, C, C++, C#, Java, Perl, Python, etc.

Syntaxe

- Il ne doit exister qu'un seul élément père par document contenant tous les autres : un élément racine.
- Structure de base : paire clef-valeur (key-value)=> "title": "The Social network"
- Les séparateurs utilisés entre deux paires/valeurs sont des virgules.
- On distingue les valeurs atomiques et les valeurs complexes (construites)
- Valeurs atomiques : chaînes de caractères, nombres (entiers, flottants) et valeurs booléennes (true ou false)
- Valeurs complexes 1 : les objets=> Un objet est un ensemble de paires clef-valeur. Au sein d'un ensemble de paires, une clef apparait au plus une fois (NB : les types de valeurs peuvent être distincts).
- Un objet peut être utilisé comme valeur (dite complexe) dans une paire clef-valeur=> Un objet JSON peut contenir d'autres objets JSON.
- Valeurs complexes 2 : les tableaux=> Un tableau (array) est une liste de valeurs (dont le type n'est pas forcément le même).
- Imbrication sans limite: tableaux de tableaux, tableaux d'objets contenant eux-mêmes des tableaux, etc.
- Un document est un objet. Il peut être défini par des objets et tableaux imbriqués autant de fois que nécessaire.
- Tout fichier JSON bien formé doit être soit un objet (commençant par "{" et se terminant par "}"), soit un tableau (commençant par "[" et

terminant par "]"). Cependant ils peuvent être vides, ainsi "[]" et "{}" sont des JSON valides.

Exemples:

Objet

```
{
   "nom" : "Bendali",
   "prenom" : "Ali",
   "age" : "73",
   "Adresse" : "Constanine"
}
```

Tableau

```
[ {
    "id":7,
    "nom":"Bendahi",
    "prenom": "mohamed",
    "sexe": "homme",
    "date_naissanse": "10/10/1991",
    "telephone": "066666666",
    "adresse":"Biskra",
    "email":"bmohamed1991@gmail.com",
}]
```

JSON vs XML:

- JSON plus léger et intuitif que XML=> il peut prendre bien moins de place que XML pour représenter un jeu de données équivalent (Cela assure par exemple une **réception plus rapide** d'un large jeu de données JSON lorsque l'on fait appel à un web service que si les données étaient au format XML).

- Facile à parser pour n’importe quel langage de programmation=> La syntaxe orientée objet de JSON permet aux langages qui le parsent de facilement créer des objets représentant l'objet JSON reçu dans le langage courant.

- JSON n’a pas (encore) de langage de spécification de schéma associé=> JSON ne dispose pas du système de schémas de XML, utilisés afin de valider des fichiers XML devant respecter une certaine structure. Cela complexifie la validation de fichiers JSON massifs qui pourraient

éventuellement contenir des erreurs (nom d'un attribut mal orthographié, attribut manquant mais nécessaire, etc...).

- JSON n’a pas (encore) de langage de requête associé.

- Les langages à balises tels que XML sont particulièrement adaptés à la rédaction de documents, notamment afin de gérer efficacement la récursivité des balises. JSON n'est pas adapté à une telle utilisation, mais bien plus à la représentation de données.
- JSON ne dispose pas d'un système de traitement équivalent à XSLT, pour transformer des données et les exporter sous différents formats.

On peut donc conclure que le choix de l'un ou l'autre langage dépend du cas d'utilisation :

- JSON est plus adapté pour le partage de données formatées, mais est limité dans les types de données possibles.
- XML est plus adapté pour la création de documents, et permet de stocker n'importe quel type de données, ce qui le rend plus flexible, mais potentiellement moins lisible.