

# Présentation générale de la solution

Dans le cadre du chapitre précédent, nous avons vu comment mettre en œuvre une solution de stockage distant basé sur les technologies suivantes :

- HTML/JavaScript,
- Ajax via XMLHttpRequest (objet JavaScript ou ActiveX permettant de récupérer depuis un serveur distant des données au format XML, JSON, mais aussi HTML, ou encore du texte simple à l'aide de requêtes HTTP),
- Protocole SOAP encapsulant un en-tête et les données au format XML,
- PHP (langage orienté serveur) pour interroger la base de données,
- NuSOAP (librairie PHP dédiée à la préparation des messages SOAP),
- MySQL (serveur de base de données),
- Parser DOM pour décoder le message SOAP reçu en retour du serveur.

Les outils utilisés sont relativement génériques et vous pouvez extrapoler assez facilement en employant par exemple du côté serveur un autre SGBD (PostgreSQL, Microsoft SQL Server...), un autre langage de programmation (PHP, Microsoft ASP.NET...).

La principale difficulté serait sans doute de trouver dans notre cas une alternative à la librairie NuSOAP.

Dans le cadre de ce nouveau chapitre, nous allons redévelopper l'exemple du chapitre précédent mais en remplaçant le format de données XML par un autre format : JSON (*JavaScript Object Notation*).

JSON est un format léger d'échange de données. Il a l'avantage d'être indépendant des langages de programmation et présente la particularité d'être nativement reconnu par JavaScript. Dans les applications d'échange de données client-serveur, il est fréquemment utilisé comme alternative à XML. L'avantage est que le codage du message côté serveur est simple à réaliser et côté client il n'est pas nécessaire d'avoir recours à un parser DOM.

Bien que nous allons découvrir JSON au cours de plusieurs mini-applications dans le cadre de ce chapitre, voyons quand même l'allure générale de ce format au travers d'un exemple.

Vous trouvez ci-après un même jeu de données présenté au format JSON et au format XML :

Format JSON :

```
[
  {
    "menu": "Fichier",
    "commandes": [
      { "libelle": "Nouveau", "action": "creerDocument" },
      { "libelle": "Ouvrir", "action": "ouvrirDocument" },
      { "libelle": "Fermer", "action": "fermerDocument" }
    ]
  }
]
```

Format XML :

```
<?xml version="1.0" ?>
<barremenu>
  <menu nom="Fichier">
    <commande libelle="Nouveau " action="creerDocument" />
    <commande libelle="Ouvrir" action="ouvrirDocument" />
    <commande libelle="Fermer " action="fermerDocument" />
  </menu>
</barremenu>
```