	Documentation	<i>Date :</i> 21/02/2005
	Spécification du XMLRPC	<i>Page :</i> 1 / 6
		<i>Révision :</i> 1.01

Table des matières

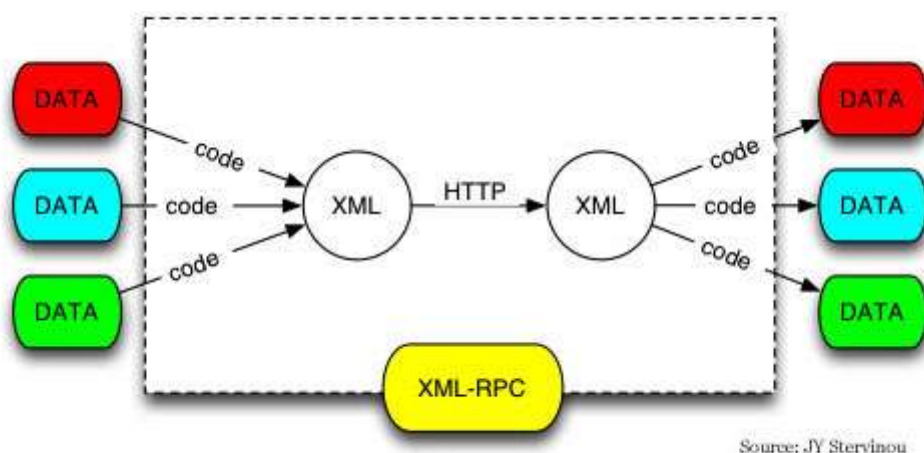
Introduction.....	1
La requête.....	1
Exemple.....	2
Entête de la requête.....	2
Format du contenu.....	2
Les scalaires <value>.....	3
Les structures <struct>.....	3
Les tableaux <array>.....	3
La réponse.....	4
Format de la réponse.....	4
La réponse en erreur.....	5
Annexe.....	6

Introduction.

Le XMLRPC est un protocole d'appel de procédure distante qui fonctionne au travers d'internet, intranet ou extranet.

Un message est une requête de type HTTP POST. Le corps de ce message est en XML. Le message est reçu et traité sur le serveur et celui ci retourne des valeurs qui sont également retournés au format XML.


Les paramètres d'une procédure peuvent être de type scalaire, numérique, chaînes de caractères, dates, etc... (Voir le tableau plus loin).



Ci dessus le schéma représentatif du XMLRPC.

La requête.

L'exemple ci après effectue une demande d'ajout d'un nouveau produit, en retour nous obtiendrons le prochain code produit alloué en fonction du paramétrage, la valeur du code barre.

	Documentation	<i>Date :</i> 21/02/2005
	Spécification du XMLRPC	<i>Page :</i> 2 / 6
		<i>Révision :</i> 1.01

Exemple.

```

POST /RPC2 HTTP/1.0
User-Agent: opengescom (Linux)
Host: www.opengescom.org
Content-Type: text/xml
Content-length: 181

<?xml version="1.0"?>
<methodCall>
  <methodName>opgc.requestNewPro</methodName>
  <params>
    <param>
      <value><i4>1</i4></value>
    </param>
  </params>
</methodCall>

```

Entête de la requête.

Le format de l'**URI** dans la première ligne de l'entête n'est pas spécifié, par exemple il peut être vide, juste un / , si le serveur manipule seulement des appels **XMLRPC**. Néanmoins si le serveur manipule à la fois une multitude de type de requête **HTTP**, la présence de l'**URI** nous permet de router la requête **XMLRPC** vers le bon système d'écoute.

Le « **User-Agent** » et « **Host** » doivent être spécifiés.

Le « **Content-Type** » doit être « **text/xml** »

Le « **Content-Length** » doit être spécifié et correct.


Format du contenu.

Le payload est en XML, et contient une seule balise **<methodCall>**.

La balise **<methodCall>** doit contenir une seule sous balise **<methodName>**, la valeur contenue entre la balise **<methodName>** est de type chaîne de caractères et contient le nom de la méthode à appeler. La méthode peut être composée de majuscule, minuscule des lettres de A-Z, des chiffres de 0-9 et autres caractères comme le point(.) le pipe (|) et le slash (/). Il appartient entièrement au serveur de décider comment interpréter les caractères de la méthode.

Par exemple, la méthode peut être le nom d'un fichier qui contient un script que l'on exécute lorsque l'on effectue une requête. Cela peut être également le nom d'une colonne dans une base de données.

Si la requête fait appel à des paramètres, la balise **<methodCall>** doit contenir une seule balise **<params>**, celle-ci pourra contenir une ou plusieurs balises **<param>** (attention il n'y a pas de « s ») dont chacune est composée d'une balise **<value>**.

	Documentation	<i>Date :</i> 21/02/2005
	Spécification du XMLRPC	<i>Page :</i> 3 / 6
		<i>Révision :</i> 1.01

Les scalaires <value>.

Les valeurs peuvent être des grandeurs scalaires, le type est indiqué en intégrant la valeur à l'intérieur des étiquettes énumérées dans ce tableau.

<i4> ou <int>	Entier signé de 4 bits	-5432
<boolean>	(0) false et (1) true	1
<string>	Chaîne de caractères	Bonjour KrysKool
<double>	nombre signé à double précision à virgule flottante	-12.5432
<dateTime.iso8601>	Date et heure au format ISO 8601	19980717T14:08:55
<base64>	Encodage binaire en base64	eW91IGNhbid0IHJIYWQg=

Si le type n'est pas indiqué, le type **string** sera utilisé par défaut.

Les structures <struct>.

Une valeur peut être de type <struct>.

Une balise <struct> contient une ou plusieurs balises <member> et chaque balise <member> contient une balise <name> et <value>.

Ci-dessous un exemple de balise <struct>.


```
<struct>
  <member>
    <name>minimum</name>
    <value><i4>18</i4></value>
  </member>
  <member>
    <name>maximum</name>
    <value><i4>139</i4></value>
  </member>
</struct>
```

Une balise <struct> peut être récursive. Chaque balise <value> peut contenir une balise <struct> ou d'autres types, incluant les types <array> décrits ci-dessous.

Les tableaux <array>.

Une valeur peut être du type <array>.

Une balise <array> contient un seul élément <data>, et cet élément <data> contient un ou plusieurs éléments <value>.

	Documentation	<i>Date :</i> 21/02/2005
	Spécification du XMLRPC	<i>Page :</i> 4 / 6
		<i>Révision :</i> 1.01

```
<array>
  <data>
    <value><i4>12</i4></value>
    <value><string>Alencon</string></value>
    <value><boolean>0</boolean></value>
    <value><i4>-31</i4></value>
  </data>
</array>
```

Les éléments `<array>` n'ont pas des noms.

Vous pouvez mélanger des types comme l'exemple ci-dessus l'illustre.

Les balises `<array>` peuvent être récursives, chaque élément `<value>` peut contenir un élément `<array>` ou tous autres types, y compris l'élément `<struct>` comme décrit précédemment.

La réponse.


Ci dessous la réponse à la requête XMLRPC précédente.

```
HTTP/1.1 200 OK
Connection: close
Content-Length: 158
Content-Type: text/xml
Date: Tue, 22 Feb 2005 18:30:08 GMT
Server: OpenGesCom/1.0.1-Linux Debian

<?xml version="1.0"?>
<methodResponse>
  <params>
    <param>
      <struct>
        <member>
          <name>code produit</name>
          <value><string>AB000010</string></value>
        </member>
        <member>
          <name>code barre</name>
          <value><string>3270190113508</string></value>
        </member>
      </struct>
    </param>
  </params>
</methodResponse>
```

Format de la réponse.

A moins qu'il y est une erreur dans la couche basse du protocole **HTTP**, la réponse retourne toujours **200 OK**.

	Documentation	<i>Date :</i> 21/02/2005
	Spécification du XMLRPC	<i>Page :</i> 5 / 6
		<i>Révision :</i> 1.01

Le « **Content-Type** » est **text/xml**.

Le « **Content-Length** » doit être renseigné et correct.

Le corps de la réponse est une structure **XML**, la balise **<methodResponse>** contient un seul élément **<params>**, qui contient une seule balise **<param>** et par conséquent un élément **<value>**.

La balise **<methodResponse>** peut également contenir un balise **<fault>**, qui contient un élément **<value>** qui inclus également une balise **<struct>** qui contient 2 éléments, un appelé **FaultCode** de type entier, et un autre **FaultString** de type caractères.

Une balise **<methodResponse>** ne peut contenir en même temps la balise **<params>** et **<fault>**.


La réponse en erreur.

```

HTTP/1.1 200 OK
Connection: close
Content-Length: 426
Content-Type: text/xml
Date: Tue, 22 Feb 2005 18:30:08 GMT
Server: OpenGesCom/1.0.1-Linux Debian

<?xml version="1.0"?>
<methodResponse>
  <fault>
    <value>
      <struct>
        <member>
          <name>faultCode</name>
          <value><int>4</int></value>
        </member>
        <member>
          <name>faultString</name>
          <value><string>Too many parameters.</string></value>
        </member>
      </struct>
    </value>
  </fault>
</methodResponse>

```

	Documentation	<i>Date :</i> 21/02/2005
	Spécification du XMLRPC	<i>Page :</i> 6 / 6
		<i>Révision :</i> 1.01

Annexe.

Cette documentation a été réalisé à l'aide des spécifications écrites sur le site <http://www.xmlrpc.com/>

Copyright and disclaimer

© Copyright 1998-2003 UserLand Software. All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and these paragraphs are included on all such copies and derivative works.

This document may not be modified in any way, such as by removing the copyright notice or references to UserLand or other organizations. Further, while these copyright restrictions apply to the written XML-RPC specification, no claim of ownership is made by UserLand to the protocol it describes. Any party may, for commercial or non-commercial purposes, implement this protocol without royalty or license fee to UserLand. The limited permissions granted herein are perpetual and will not be revoked by UserLand or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and USERLAND DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

<http://www.opengescom.org/>