

Utilisation du webservice SQL

Si vous voulez utiliser le web service pour accéder à vos données Logistics, vous devez impérativement passer par [l'Installation du Webservice SQL](#).



Utilitaire de test

Vous pouvez consulter la section [Test du Webservice SQL](#) pour obtenir un utilitaire de test du webservice

Le webservice de Logistics repose sur une requête principale dont le premier paramètre correspond à la fonction qui est appelée.

Voici le descriptif de cette fonction :

`wsrequest (Fonction, Path, Key, Lang, XmlHeader, XmlFile, XmlFooter)`

Paramètre	Description
Fonction	Nom de la fonction appelée
Path	Paramètre historique qui n'est plus utilisé dans le cadre des procédures standard du webservice mais peut être utilisé dans les procédures ajoutées en plug in
Key	Clé applicative
Lang	Paramètre utilisé dans le cas la procédure WSGETARTADDITIONALFIELDS ou dans les procédures ajoutées en plug in dans les autres cas celle-ci est optionnelle
XmlHeader	1er xml de paramétrage, voir le descriptif de la procédure appelée
XmlFile	2eme xml de paramétrage, voir le descriptif de la procédure appelée
XmlFooter	3eme xml de paramétrage, voir le descriptif de la procédure appelée

Le web service est constitué de plusieurs fonction décrites ci-dessous.

Ces méthodes permettent par exemple de créer, modifier un utilisateur ou encore d'avoir le prix actuel d'un article donné pour un stock bien précis, etc...

Sur cette page:

- [Connexion au webservice](#)
- [Projets exemples](#)
- [WsLogin](#)
- [WsGetLogin](#)
- [WsManageCustomer](#)
- [WsGetCustInfo](#)
- [WsGetPrice_Stock](#)
- [WsGetDeliveryAdress](#)
- [WsGetContacts](#)
- [WsManageContact](#)
- [WsCheckOrder - WsPutOrder](#)
- [WsGetArtInfo](#)
- [WsGetPasswordFromEmail](#)
- [WsArtGetItems](#)
- [WsArtGetArtIdComp](#)
- [WsGetPriceArray](#)
- [WsGetStringQuery](#)
- [GetArticles](#)
- [GetAnalyticsTree](#)
- [GetThirdLogin](#)
- [CheckDocument - AddDocument](#)

Connexion au webservice

La connexion au webservice est basé sur une clé partagée entre le webservice et l'application utilisatrice.

Il suffit donc de remplir le paramètre Key de la méthode wsrequest avec la valeur de la clé partagée



Veillez consulter la [configuration de Logistics pour le module webservice](#) pour plus de détail sur la configuration de cette clé partagée.



La requête **WsLogin** est présente principalement pour une question historique mais permet de récupérer la clé applicative en spécifiant un code utilisateur et son mot de passe.

wsloginrequest (PcCtrl)

PcCtrl

```
<?xml version = '1.0' encoding='Windows-1252' standalone='yes'?>
<VFPData>
  <login>
    <userid>WSUSER</userid>
    <pswd>Password_WSUSER</pswd>
    <compid>FOLDER</compid>
    <version></version>
    <PrgRoot></PrgRoot>
  </login>
</VFPData>
```

Projets exemples

les projets présentés sont des exemples d'utilisation du webservice dans divers langage de programmation et n'ont pas pour but d'être directement installer en production.

- [PHP - Site web Ecommerce](#)
- [C# - Exemple d'utilisation des requêtes WSGETSTRINGQUERY et WSGETPRICE_STOCK](#) (Projet en visual studio 2017)

WsLogin

Cette fonction `wsLogin` renvoie trois paramètres (`Path`, `Key`, `language`). Vous devez ensuite utiliser ces 3 paramètres pour faire appel aux autres fonctions du webservice.

Exemple en ASP

```

private function ConnectEdb
LsUSER = "WSUSER" // Utilisateur Logistics
LsPSWD = "987654" // Mot de passé de l'utilisateur Logistics
LsCOMPANY = "DEMO" // Nom du dossier Logistics
LsVERSION = "109" // Version du webservice
LsPRGROOT = "\\server\winbooks\data\logistics\" // Chemin où se trouve le plugin (utilisé par le web service).
Attention, comme le web service peut se trouver sur un autre serveur que celui des pages du site, il faut
impérativement lui envoyer un chemin UNC. Le chemin dans le fichier Logistics.ini doit également être un chemin
UNC.

LsCTRL = "<?xml version = '1.0' encoding='Windows-1252' standalone='yes'?>" & _
        "<VFPData><login>" & _
        "<userid>" & LsUSER & "</userid>" & _
        "<pswd>" & LsPSWD & "</pswd>" & _
        "<compid>" & LsCOMPANY & "</compid>" & _
        "<version>" & LsVERSION & "</version>" & _
        "<PrgRoot>" & LsPRGROOT & "</PrgRoot></login></VFPData>"

set soapclient = CreateObject("MSSOAP.SoapClient30")
soapclient.ClientProperty("ServerHttpRequest") = True
soapclient.mssoapinit(WSPath)

Lsresult = soapclient.login (LsCTRL)
set oLogin = new clsLogin
oLogin.XmlLoginResponse = LsResult

session("edb_Path") = oLogin.DbPath
session("edb_Key") = oLogin.DbKey
session("edb_Lang") = oLogin.Language

```

Exemple en C#

```

String lsValue = "";
String lsName = "";
String LsCTRL = "<?xml version = '1.0' encoding='Windows-1252' standalone='yes'?>" +
        "<VFPData><login>" +
        "<userid>" + psUSER + "</userid>" +
        "<pswd>" + psPSWD + "</pswd>" +
        "<compid>" + psCOMPANY + "</compid>" +
        "<version>" + psVERSION + "</version>" +
        "<PrgRoot>" + psPRGROOT + "</PrgRoot></login></VFPData>";
WsFlex ws = new WsFlex("");
psResultLogin = ws.LOGIN(LsCTRL);

```

En C#, **WsFlex** doit être déclaré dans les WebRéférences de votre projet qui pointe vers le fichier WSDL de votre webservice.

Exemple en PHP

```

<?php

// 1. Initialisation

$client = new SoapClient("http://217.112.181.34/wsflexDotNet/ServiceFlexDotNet.asmx?wsdl", array('soap_version'
=> SOAP_1_2));
$LsUSER = "WSUSER"; // Utilisateur Logistics
$LsPSWD = "987654"; // Mot de passé de l'utilisateur Logistics
$LsCOMPANY = "DEMO"; // Nom du dossier Logistics
$LsVERSION = "109"; // Version du webservice
$LsPRGROOT = "\\server\winbooks\data\logistics\\";
$LsCTRL = "<?xml version = '1.0' encoding='Windows-1252' standalone='yes' ?>
<VFPData>
  <login>
    <userid>".$LsUSER."</userid>
    <pswd>".$LsPSWD."</pswd>
    <compid>".$LsCOMPANY."</compid>
    <version>".$LsVERSION. "</version>
    <PrgRoot>".$LsPRGROOT."</PrgRoot>
  </login>
</VFPData>";

try
{
  $params = array("PcCtrl" => $LsCTRL);
  $Res = $client->__call('Login', array("parameter" => $params));

  $language = "F";
  $dbpath = $LsPRGROOT."".$LsCOMPANY."\\\\";
  $dbkey = $Res->LOGINResult;

  //echo 'RESPONSE: ' . $client->__getLastResponse()."<br>";
  //echo 'REQUEST : ' . $client->__getLastRequest()."<br>";
}
catch (SoapFault $fault)
{
  print_r($fault);
  throw new Exception($fault->getMessage());
}

// 2. Request

$xmlFile = '<?xml version="1.0" encoding="Windows-1252" standalone="yes" ?>
<VFPData>
  <xmlreturn>
    <custid>WS00000003</custid>
  </xmlreturn>
</VFPData>';

try
{
  $param = array("PcFunc" => "WsGetCustInfo", "PcPath" => $dbpath, "PcKey" => $dbkey, "PcLang" => $language,
"Pcxmlheader" => "a", "pcxmlfile" => $xmlFile, "pcxmlfooter" => "a");
  $cust = $client->__call("WsRequest", array("parameter" => $param));
  var_dump($cust);
}
catch (SoapFault $fault)
{
  print_r($fault);
  throw new Exception($fault->getMessage());
}

?>

```

[Haut de page](#)

WsGetLogin



Dans toutes les versions inférieures à la version 5.5, le mot de passe ne doit pas dépasser 20 caractères. A partir de la version 5.5, il n'y a plus de limitation du nombre de caractères dans le mot de passe.

Exemple d'XML envoyé

XMLHEADER

```
<?xml version="1.0" encoding="Windows-1252" standalone="yes" ?>
<VFPData>
  <xmlresult>
    <chkmail>jfcu@spiritsystems.be</chkmail>
    <pswd>123</pswd>
  </xmlresult>
</VFPData>
```

XMLFILE

a

XMLFOOTER

a


Exemple d'XML reçu

```
<?xml version="1.0" encoding="Windows-1252" standalone="yes" ?>
<VFPData>
  <xmlreturn>
    <name>Fauquet Sprl.</name>
    <name2>Fauquet Geoffrey</name2>
    <adrname />
    <adrname2 />
    <adrstreet />
    <adrstreet2 />
    <adrcountry />
    <adrzip />
    <adrcity />
    <adrfax />
    <adrphone />
    <adrphone2 />
    <userid />
    <email />
    <civility />
    <title />
    <customer>true</customer>
    <supplier>false</supplier>
    <groupid>WS00000001</groupid>
    <custid>P-WS00000003</custid>
    <custtype />
    <vat />
    <vatcountry />
    <discount>0</discount>
    <saledelay>0</saledelay>
    <paydelay />
    <paydisc>0</paydisc>
    <other>false</other>
    <paymode />
    <web>true</web>
    <webpsw />
    <webwhere />
    <points>0</points>
    <errornr>ERROR 0</errornr>
    <language />
    <dbkey />
    <logmessage>Consumer is correct</logmessage>
    <displayinventory>0</displayinventory>
    <u_test />
    <c_prenom>John.</c_prenom>
    <c_num>18</c_num>
    <anal />
  </xmlreturn>
</VFPData>
```

[Haut de page](#)

WsManageCustomer

Cette fonction permet d'ajouter ou de modifier un client.
Si aucun **Custid** n'est indiqué, le client sera alors créé sinon il sera modifié

 Le mot de passe ne doit pas dépasser 20 caractères.

Exemple d'appel en CSharp

```
TextBox_XmlResult.Text = ws.WsRequest("WSMANAGECUSTOMER", TextBox_XmlHeaderSend.Text, TextBox_XmlFileSend.Text,
TextBox_XmlFooterSend.Text);
```

Exemple d'XML envoyé

XMLHEADER

a

XMLFILE

```
<?xml version="1.0" encoding="Windows-1252" standalone="yes" ?>
<VFPData>
  <xmlreturn>
    <name>Fauquet Sprl</name>
    <name2>Fauquet Geoffrey</name2>
  </xmlreturn>
</VFPData>
```

XMLFOOTER

a

Exemple d'XML reçu

```
<?xml version="1.0" encoding="Windows-1252" standalone="yes" ?>
<VFPData>
  <xmlreturn>
    <custid>C-WS00000003</custid>
    <userid />
    <accountid />
    <groupid>WS00000003</groupid>
    <name>Fauquet Sprl</name>
    <name2>Fauquet Geoffrey</name2>
    <adrstreet />
    <adrzip />
    <adrcity />
    <adrphone />
    <adrphone2 />
    <adrfax />
    <language />
    <vat />
    <memo />
    <vatid />
    <s_deleted>>false</s_deleted>
    <s_creatdate>2008-02-26T10:46:07</s_creatdate>
    <s_moddate>2008-02-26T10:53:08</s_moddate>
    <custtype />
    <currid />
    <agent />
    <artcateg />
    <artcoef>0</artcoef>
    <bankname />
    <banknr />
    <paydelay />
    <paymode />
    <paydisc>0.0</paydisc>
    <credfile />
    <credamount>0</credamount>
    <crednote />
    <priority />
    <fullline>>false</fullline>
    <situation />
    <transport />
    <ca>0</ca>
    <bankparam />
    <tarif>0</tarif>
```

```
<ana1 />
<ana2 />
<adrcountry />
<stkid />
<adrname />
<adrname2 />
<title />
<supplier>>false</supplier>
<customer>>true</customer>
<extref />
<importid />
<other>>false</other>
<custaccid />
<discount>0</discount>
<currid2 />
<adrstreet2 />
<invstreet2 />
<vatcountry />
<email />
<website />
<buyvatid />
<saledelay>0</saledelay>
<s_repdate />
<blocktype>0</blocktype>
<blockwhen>0</blockwhen>
<blockmax>0</blockmax>
<codexpgrp />
<copycount>0</copycount>
<invpos>>false</invpos>
<invartgrp>>false</invartgrp>
<amtfranco>0</amtfranco>
<artport />
<vatcat />
<modified>>false</modified>
<civility />
<web>>false</web>
<webpsw />
<webwhere />
<webdispinv>0</webdispinv>
<forecolor />
<gdisc>0</gdisc>
<u_test />
<s_creuid />
<s_moduid />
</xmlreturn>
</VFPData>
```

[Haut de page](#)

WsGetCustInfo

Cette fonction permet d'avoir les informations d'un client.

Exemple en ASP

```
set soapclient = CreateObject("MSSOAP.SoapClient30")
soapclient.ClientProperty("ServerHTTPRequest") = True
soapclient.mssoapinit(WSPath)

LSresult = soapclient.wsrequest ("WSGETCUSTINFO",LsPath,LsKey,LsLang,"a",LsXMLFile,"a")
```

Exemple d'appel en CSharp

```
ws.WsRequest(PcProcName, pPath, pKey, pLang, pcXmlHeader, pcXmlFile, pcXmlFooter);
```


Exemple d'XML envoyé

XMLHEADER

a

XMLFILE

```
<?xml version="1.0" encoding="Windows-1252" standalone="yes" ?>
<VFPData>
  <xmlreturn>
    <custid>C-WS00000001</custid>
  </xmlreturn>
</VFPData>
```

XMLFOOTER

a

Exemple d'XML reçu

```
<?xml version="1.0" encoding="Windows-1252" standalone="yes" ?>
<VFPData>
  <xmlreturn>
    <custid>WS00000316</custid>
    <userid />
    <accountid />
    <groupid>WS000316</groupid>
    <name>Spirit Systems sprl</name>
    <name2>Cuylits Jean-François.</name2>
    <adrstreet>Fond Jean Paques 6C</adrstreet>
    <adrzip>1438</adrzip>
    <adrcity>Louvain-La-Neuve</adrcity>
    <adrphone>010 45 12 44</adrphone>
    <adrphone2>0475 78 82 57</adrphone2>
    <adrfax />
    <language>F</language>
    <vat>04737.109.813</vat>
    <memo />
    <vatid />
    <s_deleted>>false</s_deleted>
    <s_ccreate />
    <s_moddate />
    <custtype />
    <currid />
    <agent />
    <artcateg />
    <artcoef>0</artcoef>
    <bankname />
    <banknr />
    <paydelay />
    <paymode />
    <paydisc>0.0</paydisc>
    <credfile />
    <credamount>0</credamount>
    <crednote />
    <priority />
    <fullline>>false</fullline>
    <situation />
    <transport />
    <ca>0</ca>
    <bankparam />
    <tarif>0</tarif>
```

```
<anal />
<adrcountry>BE</adrcountry>
<stkid />
<adrname />
<adrname2 />
<title>Monsieur</title>
<supplier>>false</supplier>
<customer>>false</customer>
<extref />
<importid />
<other>>false</other>
<custaccid />
<discount>0</discount>
<currid2 />
<adrstreet2 />
<invstreet2 />
<vatcountry>BE</vatcountry>
<email>jfcu@spiritsystems.be</email>
<website />
<buyvatid />
<saledelay>0</saledelay>
<s_repdate />
<blocktype>0</blocktype>
<blockwhen>0</blockwhen>
<blockmax>0</blockmax>
<codexpgrp />
<copycount>0</copycount>
<invpos>>false</invpos>
<invartgrp>>false</invartgrp>
<vatcat />
<modified>>false</modified>
<civility />
<web>>false</web>
<webpsw>123</webpsw>
<webwhere />
<webdispinv>0</webdispinv>
<forecolor />
<gdisc>0</gdisc>
<u_test />
<s_creuid />
<s_moduid />
</xmlreturn>
</VFPData>
```

[Haut de page](#)

WsGetPrice_Stock

Cette fonction permet d'avoir le prix pour un article donné ainsi que le nombre d'article total et par stock spécifié.

Exemple d'XML envoyé

XMLHEADER

```
<?xml version="1.0" encoding="Windows-1252" standalone="yes" ?>
<VFPData>
  <Customer>
    <custid>C-0100000009</custid>
  </Customer>
</VFPData>
```

XMLFILE

```
<?xml version="1.0" encoding="Windows-1252" standalone="yes" ?>
<articles>
  <article>
    <artid>C2</artid>
    <color />
    <size />
  </article>
  <warehouses>
    <Warehouse>A</Warehouse>
    <Warehouse>B</Warehouse>
  </warehouses>
</articles>
```

XMLFOOTER

a

Exemple d'XML reçu

```
<?xml version="1.0" encoding="Windows-1252" standalone="yes" ?>
<VFPData>
  <vatinc>.T.</vatinc>
  <xmlreturnst>
    <xartid>C2</xartid>
    <xcolor />
    <xsize />
    <xwarehouse>
      <Xname>A</Xname>
      <Qty>5000</Qty>
    </xwarehouse>
    <xwarehouse>
      <Xname>B</Xname>
      <Qty>3000</Qty>
    </xwarehouse>
    <xqty>8000</xqty>
    <xprice>0</xprice>
    <xstkinday>0</xstkinday>
    <xremark>x</xremark>
  </xmlreturnst>
</VFPData>
```

[Haut de page](#)

WsGetDeliveryAdress

Cette fonction permet d'avoir les adresses de livraisons d'un client.

Exemple d'XML envoyé

XMLHEADER

a

XMLFILE

```
<?xml version="1.0" encoding="Windows-1252" standalone="yes" ?>
<VFPData>
  <xmlreturn>
    <custid>C-WS00000001</custid>
  </xmlreturn>
</VFPData>
```

XMLFOOTER

a

Exemple d'XML reçu

```
<?xml version="1.0" encoding="Windows-1252" standalone="yes" ?>
<VFPData>
  <xmlreturn>
    <name>Geo SPRL.</name>
    <adrstreet>Place du charbonnage, 2</adrstreet>
    <adrstreet2>Trazegnies</adrstreet2>
    <adrzip>6183</adrzip>
    <adrcity>Trazegnies</adrcity>
    <adrcountry>BE</adrcountry>
    <persid>P-WS00000001</persid>
  </xmlreturn>
</VFPData>
```

[Haut de page](#)

WsGetContacts

Cette fonction permet d'avoir les contacts d'un client

Exemple d'XML envoyé

XMLHEADER

a

XMLFILE

```
<?xml version="1.0" encoding="Windows-1252" standalone="yes" ?>
<VFPData>
  <xmlreturn>
    <custid>C-WS00000001</custid>
  </xmlreturn>
</VFPData>
```

XMLFOOTER

a

Exemple d'XML reçu

```
<?xml version="1.0" encoding="Windows-1252" standalone="yes" ?>
<VFPData>
  <xmlreturn>
    <name>Geo SPRL.</name>
    <adrstreet>Place du charbonnage, 2</adrstreet>
    <adrstreet2>Trazegnies</adrstreet2>
    <adrzip>6183</adrzip>
    <adrcity>Trazegnies</adrcity>
    <adrcountry>BE</adrcountry>
    <persid>P-WS00000001</persid>
  </xmlreturn>
  <xmlreturn>
    <name>Test</name>
    <adrstreet>Rue du bois</adrstreet>
    <adrstreet2>Drogenbos</adrstreet2>
    <adrzip>1000</adrzip>
    <adrcity>Bruxelles</adrcity>
    <adrcountry />
    <persid>P-WS00000002</persid>
  </xmlreturn>
</VFPData>
```

[Haut de page](#)

WsManageContact

Cette fonction permet d'ajouter ou de modifier un contact. Pour cela, il faut envoyer le **custID** du client visé. Si le **persid** est spécifié et existant, alors le contact sera modifié sinon il sera ajouté.

Exemple d'XML envoyé

XMLHEADER

a

XMLFILE

```
<?xml version="1.0" encoding="Windows-1252" standalone="yes" ?>
<VFPData>
  <xmlreturn>
    <name>Spirit Systems sprl.</name>
    <name2>Cuylits Jean-François.</name2>
    <adrstreet>Fond Jean Paques 6C</adrstreet>
    <adrstreet2 />
    <adrcountry>BE</adrcountry>
    <adrzip>1438</adrzip>
    <adrcity>Louvain-La-Neuve</adrcity>
    <adrfax />
    <adrphone>010 45 12 44</adrphone>
    <adrphone2>0475 78 82 57</adrphone2>
    <email>jfcu@spiritsystems.be</email>
    <title>Monsieur</title>
    <vat>0437.109.813</vat>
    <vatcountry>BE</vatcountry>
    <web>true</web>
    <webpsw>123</webpsw>
    <language>F</language>
    <CustId>C-WS00000001</CustId>
    <persid>P-WS00000003</persid>
  </xmlreturn>
</VFPData>
```

XMLFOOTER

a

Exemple d'XML reçu

```
<?xml version="1.0" encoding="Windows-1252" standalone="yes" ?>
<VFPData>
  <xmlreturn>
    <name>Geo SPRL.</name>
    <adrstreet>Place du charbonnage, 2</adrstreet>
    <adrstreet2>Trazegnies</adrstreet2>
    <adrzip>6183</adrzip>
    <adrcity>Trazegnies</adrcity>
    <adrcountry>BE</adrcountry>
    <persid>P-WS00000001</persid>
  </xmlreturn>
  <xmlreturn>
    <name>Test</name>
    <adrstreet>Rue du bois</adrstreet>
    <adrstreet2>Drogenbos</adrstreet2>
    <adrzip>1000</adrzip>
    <adrcity>Bruxelles</adrcity>
    <adrcountry />
    <persid>P-WS00000002</persid>
  </xmlreturn>
</VFPData>
```

[Haut de page](#)

WsCheckOrder - WsPutOrder

Cette fonction permet de voir si l'ajout d'une commande est possible. Si oui, on peut utiliser **WsPutOrder** pour créer le document. Le header contient les champs correspondants aux informations du client et le File contient le détail de la commande.

Exemple d'XML envoyé

XMLHEADER

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<order>
  <docdet>
    <thirdid>P-WS00000002</thirdid>
    <thirdname>Colin</thirdname>
    <basis1>1140490</basis1>
    <vatid1>21</vatid1>
    <vatpc1>21</vatpc1>
    <vatamt1>239503</vatamt1>
    <basis2 />
    <vatid2 />
    <vatpc2 />
    <vatamt2 />
    <basis3 />
    <vatid3 />
    <vatpc3 />
    <vatamt3 />
    <topay>1379993</topay>
  </docdet>
</order>
```

XMLFILE

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<order>
  <docdet>
    <artid>DELL1700</artid>
    <artname>Dell 1700</artname>
    <qty>1000</qty>
    <unitprice>561980</unitprice>
    <amount>561980</amount>
    <vatid>21</vatid>
    <vatpc>21</vatpc>
  </docdet>
  <docdet>
    <artid>HPLASERCOLOR3000</artid>
    <artname>HP Color LaserJet 3000</artname>
    <qty>1000</qty>
    <unitprice>578510</unitprice>
    <amount>578510</amount>
    <vatid>21</vatid>
    <vatpc>21</vatpc>
  </docdet>
</order>
```

XMLFOOTER

a

Exemple d'XML reçu

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<VFPData>
  <returncode>OK</returncode>
  <Xorderid>382</Xorderid>
  <headgroups>
    <headgroup code="L100FAC" description="Coordonnées de facturation">
      <header>
        <ncol>2</ncol>
        <col01 type="C">Libellé</col01>
        <col02 type="M">Valeur</col02>
      </header>
      <recs>
        <rec>
          <col01>Code client</col01>
          <col02>WS00000001</col02>
        </rec>
        <rec>
          <col01>Nom</col01>
          <col02>Colin</col02>
        </rec>
        <rec>
          <col01>Adresse</col01>
          <col02>Chaussée de Tubize 483<br>BE 1420 Braine l'alleud</col02>
        </rec>
      </recs>
    </headgroup>
    <headgroup code="L200VAT" description="Totaux">
      <header>
        <ncol>2</ncol>
        <col01 type="C">Libellé</col01>
        <col02 type="M">Valeur</col02>
      </header>
      <recs>
        <rec>
```

```

        <col01>Htva</col01>
        <col02>942.56</col02>
    </rec>
    <rec>
        <col01>Tva</col01>
        <col02>197.94</col02>
    </rec>
    <rec>
        <col01>A payer</col01>
        <col02>1140.50</col02>
    </rec>
    <rec>
        <col01>Devise</col01>
        <col02>EUR</col02>
    </rec>
</recs>
</headgroup>
</headgroups>
<body>
    <header>
        <ncol>7</ncol>
        <col01 type="C">code</col01>
        <col02 type="C">description</col02>
        <col03 type="N" dec="1">quantité</col03>
        <col04 type="N" dec="2">prix</col04>
        <col05 type="N" dec="2">remise</col05>
        <col06 type="N" dec="2">htva</col06>
        <col07 type="C">color</col07>
    </header>
    <recs>
        <rec>
            <col01>DELL1700</col01>
            <col02>Dell 1700</col02>
            <col03>1000</col03>
            <col04>561980</col04>
            <col05>0</col05>
            <col06>464450</col06>
            <col07 />
        </rec>
        <rec>
            <col01>HPLASERCOLOR3000</col01>
            <col02>HP Color LaserJet 3000</col02>
            <col03>1000</col03>
            <col04>578510</col04>
            <col05>0</col05>
            <col06>478110</col06>
            <col07 />
        </rec>
    </recs>
</body>
</VFPData>

```

[Haut de page](#)

WsGetArtInfo

Cette fonction permet de récupérer les informations d'un contact.

Exemple d'XML envoyé

XMLHEADER

a

XMLFILE

```
<?xml version="1.0" encoding="Windows-1252" standalone="yes" ?>
<VFPData>
  <xmlresult>
    <artid>RADIATOR</artid>
  </xmlresult>
</VFPData>
```

XMLFOOTER

a

Exemple d'XML reçu

```
<?xml version="1.0" encoding="Windows-1252" standalone="yes" ?>
<VFPData>
  <cres>
    <artid>RADIATOR</artid>
    <buyartid />
    <custid />
    <name1>RADIATOR F-11 OIL.</name1>
    <name2>RADIATOR F-11 OIL</name2>
    <family />
    <category />
    <coef>0</coef>
    <currid />
    <buyprice>0</buyprice>
    <buydate />
    <saleprice>0</saleprice>
    <stock>0</stock>
    <minqty>0</minqty>
    <maxqty>0</maxqty>
    <unit />
    <vaid>21</vaid>
    <ana1>PIECE</ana1>
    <ana2>AUTOS</ana2>
    <ana3>CITROEN</ana3>
    <ana4 />
    <memo>Information du produit</memo>
    <s_ccreate>2008-01-08T14:52:52</s_ccreate>
    <s_moddate>2008-02-07T14:56:32</s_moddate>
    <coeftarif1>0</coeftarif1>
    <saletarif1>0</saletarif1>
    <expenses>0</expenses>
    <nostock>>false</nostock>
    <artpart>>false</artpart>
    <compound>>false</compound>
    <qtypackby>0</qtypackby>
    <finalprice>0</finalprice>
    <packkg>0</packkg>
    <notinhist>>false</notinhist>
    <stocktype>1</stocktype>
    <kg>0</kg>
    <buyvaid />
    <buydisc>0</buydisc>
    <saledisc>0</saledisc>
    <artaccid />
    <actualval>0</actualval>
    <nopaydisc>>false</nopaydisc>
    <discount>0</discount>
    <buyvatpc>0</buyvatpc>
    <coeffinal>0</coeffinal>
    <pctarif1>0</pctarif1>
```

```
<pcsalepric>0</pcsalepric>
<options>>false</options>
<sleep>>false</sleep>
<buydelay>0</buydelay>
<colrange />
<s_repdate />
<buyqty>0</buyqty>
<buypack />
<saleqty>0</saleqty>
<salepack />
<linkartid />
<linkqty>0</linkqty>
<linkqtyfix>>false</linkqtyfix>
<compotype>4</compotype>
<valuetype>0</valuetype>
<proqty>0</proqty>
<web>>true</web>
<webpromo>>false</webpromo>
<webnew>>false</webnew>
<webmonth>>false</webmonth>
<webmsg1 />
<webmsg2 />
<memo1 />
<memo2 />
<webviewlev>3</webviewlev>
<webstockok>0</webstockok>
<packcost>0</packcost>
<forecolor />
<weborder>0</weborder>
<vatic2 />
<vatic3 />
<buyvatic2 />
<buyvatic3 />
<expensetyp>0</expensetyp>
<u_format />
<u_grammage />
<c_proprio />
<c_categ />
<c_format />
<c_millesim />
<c_url />
<disp>0</disp>
<c_pdv>>false</c_pdv>
<c_recomp />
<a_stock />
<s_creuid>SYSTEM</s_creuid>
<s_moduid>SYSTEM</s_moduid>
</cres>
</VFPData>
```

[Haut de page](#)

WsGetPasswordFromEmail

Cette fonction permet de récupérer un mot de passé par rapport au login (adresse email)

Exemple d'XML envoyé

XMLHEADER

a

XMLFILE

```
<?xml version="1.0" encoding="Windows-1252" standalone="yes" ?>
<VFPData>
  <xmlresult>
    <email>test@test.be</email>
  </xmlresult>
</VFPData>
```

XMLFOOTER

a

Exemple d'XML reçu

```
<?xml version="1.0" encoding="Windows-1252" standalone="yes" ?>
<VFPData>
  <curres>
    <password>123</password>
  </curres>
</VFPData>
```

[Haut de page](#)

WsArtGetItems

Cette fonction renvoie toutes les informations des articles cochés exportables sur le Web.
Ceci est bien sur un exemple, vous pouvez modifier la condition WEB = .T. avec la condition que vous cherchez à avoir.

Exemple d'XML envoyé

XMLHEADER

a

XMLFILE

```
<?xml version="1.0" encoding="Windows-1252" standalone="yes" ?>
<VFPData>
  <sql>
    <pccond>WEB = .T.</pccond>
    <pnFrom>0</pnFrom>
    <pnCount>0</pnCount>
  </sql>
</VFPData>
```

XMLFOOTER

a

Exemple d'XML reçu

```
<?xml version="1.0" encoding="Windows-1252" standalone="yes" ?>
<VFPData>
  <cures>
    <artid>02W.MC.REMUS</artid>
    <buyartid />
    <custid>1000000002</custid>
    <name1>Jeans Remus</name1>
    <name2>Jeans Remus</name2>
    <family />
    <category>28</category>
    <coef>1.40</coef>
    <currid />
    <buyprice>24.15</buyprice>
    <buyprice1>24.15</buyprice1>
    <buyprice2>24.15</buyprice2>
    <buyprice3>24.15</buyprice3>
    ...
  </cures>
</VFPData>
```

[Haut de page](#)

WsArtGetArtIdComp

Cette fonction renvoie tous les ArtId des composant d'un article.

Exemple d'XML envoyé

XMLHEADER

a

XMLFILE

```
<?xml version="1.0" encoding="Windows-1252" standalone="yes" ?>
<VFPData>
  <artid>DELL1700</artid>
</VFPData>
```

XMLFOOTER

a

Exemple d'XML reçu

```
<?xml version="1.0" encoding="Windows-1252" standalone="yes" ?>
<VFPData>
  <cres>
    <artpartid>MAXHD500</artpartid>
  </cres>
</VFPData>
```

[Haut de page](#)

WsGetPriceArray

Cette fonction renvoie les prix spécifiques pour un client et un article donné sous forme de tableau.

Dans le cas d'un prix spécifique, vous pouvez spécifier le type de prix. Ici, le 4 représente les prix spécifiques par client. Les 1 sont les prix par catégorie, 2 Prix d'achats...

Si le QtyMax est égale à 0, cela indique que le Maximum est l'infini.

Exemple d'XML envoyé

XMLHEADER

a

XMLFILE

```
<?xml version="1.0" encoding="Windows-1252" standalone="yes" ?>
<VFPData>
  <artid>TEST</artid>
  <custid>C-0100000001</custid>
</VFPData>
```

XMLFOOTER

a

Exemple d'XML reçu

```
<?xml version="1.0" encoding="Windows-1252" standalone="yes" ?>
<VFPData>
  <curres>
    <qtymin>0.00</qtymin>
    <qtymax>3.00</qtymax>
    <price>20.83</price>
  </curres>
  <curres>
    <qtymin>3.00</qtymin>
    <qtymax>6.00</qtymax>
    <price>19.23</price>
  </curres>
  <curres>
    <qtymin>6.00</qtymin>
    <qtymax>10.00</qtymax>
    <price>18.38</price>
  </curres>
  <curres>
    <qtymin>10.00</qtymin>
    <qtymax>0.00</qtymax>
    <price>17.12</price>
  </curres>
  <curres>
    <qtymin>0.00</qtymin>
    <qtymax>0.00</qtymax>
    <price>12.50</price>
  </curres>
</VFPData>
```

[Haut de page](#)

WsGetStringQuery

Vous envoyez une requête dans le tag <query> et indiquez le nom des tables (tag <table>) à ouvrir pour que celle-ci fonctionne, la fonction vous exécute la requête.

Exemple d'XML envoyé

XMLHEADER

a

XMLFILE

```
<?xml version="1.0" encoding="Windows-1252" standalone="yes" ?>
<VFPData>
  <query>SELECT NAME1 FROM ART</query>
  <table>art</table>
</VFPData>
```

XMLFOOTER

a

Exemple d'XML reçu

```
<?xml version="1.0" encoding="Windows-1252" standalone="yes" ?>
<VFPData>
  <curres>
    <name1>CORRECTION PRIX</name1>
  </curres>
  <curres>
    <name1>MOLDED,VELCRO,ANCHORS 25 MM X 25 M</name1>
  </curres>
  <curres>
    <name1>PARTICIPATION FORFAITAIRE AUX FRAIS DE TRANSPORTS</name1>
  </curres>
  <curres>
    <name1>FRAIS DE TRANSPORT 5KGS</name1>
  </curres>
  <curres>
    <name1>PARTICIPATION AUX FRAIS D'IMPRESSION TESA NET</name1>
  </curres>
</VFPData>
```

[Haut de page](#)

GetArticles

Cette requête permet de récupérer des informations articles.

il est possible de récupérer les informations suivant l'id de l'article, ou via des conditions (voir les deux types de Header)

Exemple d'XML envoyé

XMLHEADER (pour reprendre un article spécifique)

```
<articles><art id="CANON-A30"/></articles>
```

XMLHEADER (pour faire une sélection d'article)

```
<articles from="15" count="100" getcount="yes" where="web=1" order="name1" />
```

XMLFILE

```
<options>
  <fields>
    <field name="name1"/>
    <field name="name2"/>
  </fields>
  <price customer="S000010059" jnl="2-BCC"/>
  <stock allwarehouses="true" />
</options>
```

XMLFOOTER

a

Exemple d'XML reçu

```
<?xml version="1.0" encoding="Windows-1252" standalone="yes" ?>
<xml>
  <articles count="1" pricevatinclue="False">
    <art id="CANON-A30" usestock="True">
      <fields>
        <name1>Appareil photo CANON A 30</name1>
        <name2>Fotoapparaat CANON A 30</name2>
      </fields>
      <stock inventory="94" deliverable="94" />
      <prices>
        <price unitprice="531,5" discountpc="0" amount="531,5" />
      </prices>
    </art>
  </articles>
</xml>
```

[Haut de page](#)

GetAnalyticsTree

Cette requête vous retourne les analytiques article du dossier

Exemple d'XML envoyé

XMLHEADER

```
<header type = "ART" />
```

XMLFILE

a

XMLFOOTER

a

Exemple d'XML reçu

```

<analytics>
  <analytic level="1" code="APH" name="Appareil photo">
    <analytic level="2" code="CANON" name="CANON">
      <analytic level="3" code="A" name="A" />
      <analytic level="3" code="EOS" name="EOS" />
      <analytic level="3" code="G" name="G" />
      <analytic level="3" code="IXUS" name="IXUS" />
      <analytic level="3" code="PS" name="PowerShot" />
      <analytic level="3" code="S" name="S" />
      <analytic level="3" code="SX" name="SX" />
    </analytic>
    <analytic level="2" code="HP" name="HP">
      <analytic level="3" code="C" name="C" />
      <analytic level="3" code="M" name="M" />
      <analytic level="3" code="R" name="R" />
    </analytic>
    <analytic level="2" code="OLYMPUS" name="OLYMPUS">
      <analytic level="3" code="C" name="C" />
      <analytic level="3" code="E" name="E" />
      <analytic level="3" code="FE" name="FE" />
      <analytic level="3" code="SP" name="SP" />
    </analytic>
    <analytic level="2" code="SONY" name="SONY">
      <analytic level="3" code="DSC" name="DSC" />
      <analytic level="3" code="MVC" name="MVC" />
      <analytic level="3" code="RX" name="RX" />
    </analytic>
  </analytic>
  <analytic level="1" code="CAM" name="Caméra">
    <analytic level="2" code="GORPO" name="GO PRO">
      <analytic level="3" code="H3" name="HERO3" />
      <analytic level="3" code="H4" name="HERO4" />
    </analytic>
  </analytic>
  <analytic level="1" code="GSM" name="Téléphone portable">
    <analytic level="2" code="MOTOROLA" name="MOTOROLA">
      <analytic level="3" code="C" name="C" />
      <analytic level="3" code="RAZR" name="RAZR" />
      <analytic level="3" code="V" name="V" />
    </analytic>
    <analytic level="2" code="NOKIA" name="NOKIA">
      <analytic level="3" code="E" name="E" />
      <analytic level="3" code="N" name="N" />
    </analytic>
    <analytic level="2" code="SAMSUNG" name="SAMSUNG">
      <analytic level="3" code="ALPH" name="ALPHA" />
      <analytic level="3" code="D" name="D" />
      <analytic level="3" code="E" name="E" />
      <analytic level="3" code="GALA" name="GALAXY" />
      <analytic level="3" code="X" name="X" />
    </analytic>
  </analytic>
  <analytic level="1" code="NAV" name="Navigateur GPS">
    <analytic level="2" code="NAVMAN" name="NAVMAN" />
    <analytic level="2" code="TOMTOM" name="TOMTOM" />
  </analytic>
</analytics>

```

[Haut de page](#)

GetThirdLogin

Cette requête permet tester la validité un login client.
Attention le mot de passe doit être envoyé de manière chiffrée

Exemple d'XML envoyé

XMLHEADER (pour reprendre un article spécifique)

```
<header mail = "kf@winbooks.be" password="E10ADC3949BA59ABBE56E057F20F883E" />
```

XMLFILE

a

XMLFOOTER

a

Exemple d'XML reçu

```
<login status="valid" thirdid="S000010059" />
```

[Haut de page](#)

CheckDocument - AddDocument

la requête CheckDocument permet de vérifier un document avant son écriture dans la base de données via la requête AddDocument.

Les deux requêtes partagent les mêmes paramètres et retournent les mêmes valeurs, la différence est que la requête retourne toujours un nombre -1, dans le cas du AddDocument le numéro du document attribué est renvoyé.

Attention seuls les documents qui ne mouvementent pas le stock peuvent être utilisés

Exemple d'XML envoyé

XMLHEADER (pour reprendre un article spécifique)

```
<document>  
  <header thirdid="S000010059" jnl="2-BCC" />  
  <articles>  
    <art id="CANON-A30" qty="5.0" unitprice="341" discount="3" />  
  </articles>  
</document>
```

XMLFILE

a

XMLFOOTER

a

Exemple d'XML reçu

```

<doc jnl="2-BCC" number="-1">
  <header>
    <third id="S000010059">
      <groupid value="S000010059" />
      <civility value="" />
      <name value="Winbooks" />
      <address value="" />
      <address2 value="" />
      <zip value="" />
      <city value="" />
      <country value="" />
      <vat value="" />
    </third>
    <delivery type="third" id="S000010059">
      <name value="Winbooks" />
      <address value="" />
      <address2 value="" />
      <zip value="" />
      <city value="" />
      <country value="" />
      <vat value="" />
    </delivery>
    <amount totalbasis="1653,85" totalvat="347,31">
      <basis vatid="21" vatpc="21" vatamount="347,31" amount="1653,85" />
    </amount>
    <fields>
      <field id="JNL" value="2-BCC" />
      <field id="TYPE" value="CO" />
      <field id="DATE" value="03/04/2020 10:59:03" />
      <field id="THIRDID" value="S000010059" />
      <field id="THIRDGROUP" value="S000010059" />
      <field id="THIRDNAME" value="Winbooks" />
      <field id="THIRDNAME2" value="" />
      <field id="THIRDADDR" value="" />
      <field id="THIRDADDR2" value="" />
      <field id="THIRDZIP" value="" />
      <field id="THIRDCITY" value="" />
      <field id="THIRDCOUNT" value="" />
      <field id="THIRDTEL" value="" />
      <field id="THIRDFAX" value="" />
      <field id="PRIORITY" value="" />
      <field id="THIRDVAT" value="" />
      <field id="AGENT" value="" />
      <field id="PAYDISC" value="0,0" />
      <field id="PAYDELAY" value="" />
      <field id="PAYMODE" value="" />
      <field id="BASIS1" value="1653,85" />
      <field id="VATAMT1" value="347,31" />
      <field id="VATID1" value="21" />
      <field id="VATPC1" value="21" />
      <field id="BASIS2" value="0" />
      <field id="VATAMT2" value="0" />
      <field id="VATID2" value="" />
      <field id="VATPC2" value="0" />
      <field id="BASIS3" value="0" />
      <field id="VATAMT3" value="0" />
      <field id="VATID3" value="" />
      <field id="VATPC3" value="0" />
      <field id="AMOUNT" value="1653,85" />
      <field id="TOPAY" value="2001,16" />
      <field id="NUMBER" value="-1" />
    </fields>
  </header>
  <details>
    <det artid="CANON-A30" artname="Appareil photo CANON A 30" vatid="21" vatpc="21" amount="
1653,85">
      <qty total="5" unitprice="341" />
      <fields>
        <field id="QTY" value="5" />
        <field id="UNITPRICE" value="341" />
        <field id="DISCOUNT" value="3" />
      </fields>
    </det>
  </details>
</doc>

```

```
<field id="ARTID" value="CANON-A30" />
<field id="ARTNAME" value="Appareil photo CANON A 30" />
<field id="JNL" value="2-BCC" />
<field id="NUMBER" value="" />
<field id="THIRDID" value="S000010059" />
<field id="VATPC" value="21" />
<field id="VATID" value="21" />
<field id="AMOUNT" value="1653,85" />
<field id="AMTVATINC" value="2001,16" />
</fields>
</det>
</details>
</doc>
```

[Haut de page](#)