

Web services

Présentation du web service

Le web service permet de bénéficier des commandes mapping dans votre propre application. Ainsi, plus besoin de se connecter sur l'interface web de Mapping mais directement dans votre application qu'elle soit javascript, c# ou java, ...

Le web service propose les différentes actions de base disponibles sur le spooler.

Actions sur un spool :

- libérer
- suspendre
- transférer
- supprimer
- visualiser le contenu
- afficher la log
- afficher les informations

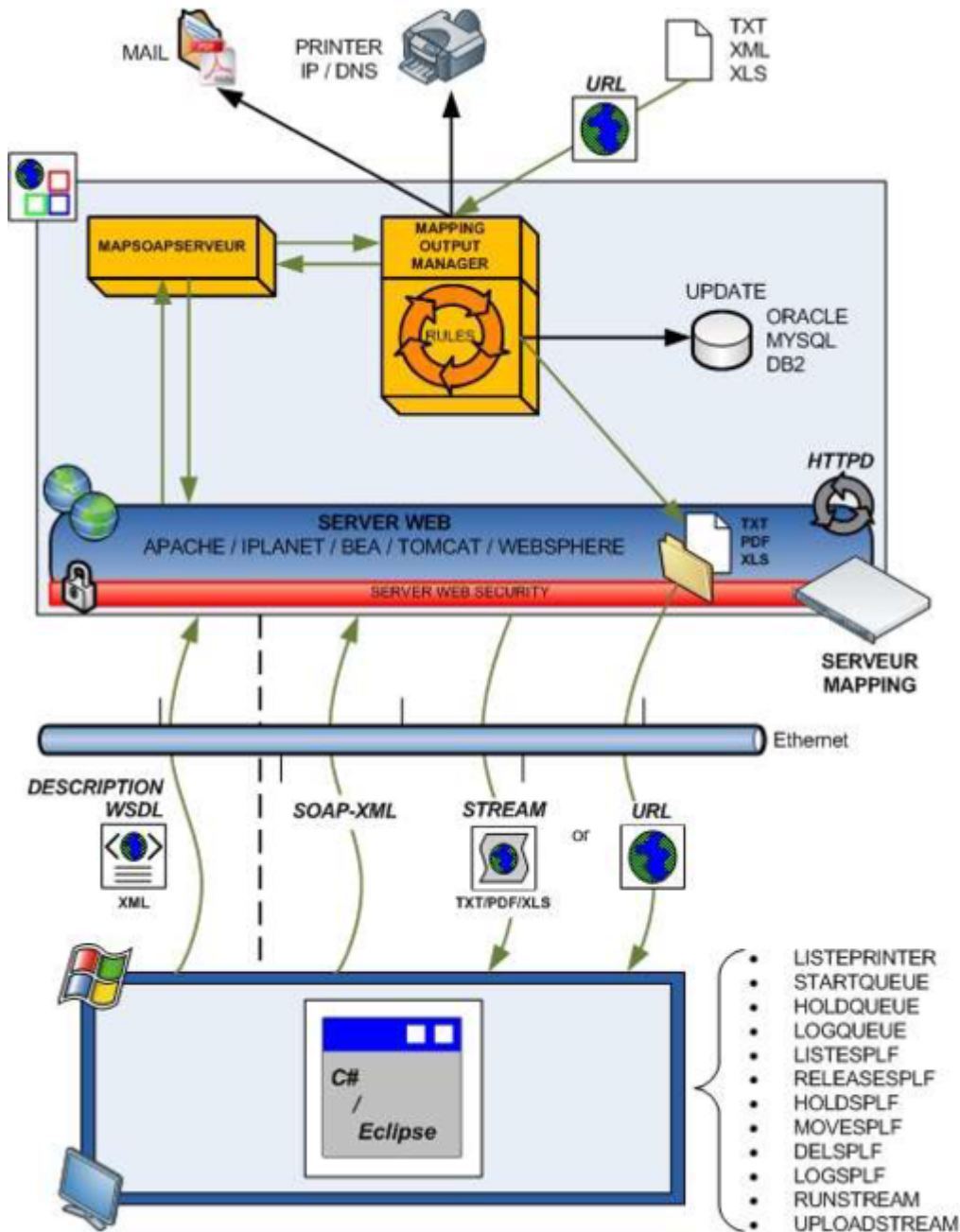
Actions sur une file d'attente :

- démarrer
- arrêter
- afficher la log
- afficher les informations

En plus des actions présentées ci dessus, le web service gère l'envoi, la prévisualisation et le remaquetage de spools. Les méthodes upload (envoi d'un spool), preview (prévisualisation d'un spool) et run (remaquetage d'un spool) proposent deux manières pour envoyer un spool ou visualiser le résultat du spool généré ou envoyé.

Pour les fichiers de taille importante, une version URL est disponible pour chaque méthode. Cette version sauvegarde le spool sur disque et transmet l'URL pour que l'application cliente ou serveur puisse lire le spool indirectement.

Dans la version stream, le contenu du spool est envoyé directement à travers le réseau. Une exception sera levée si le spool possède un volume de données trop important. Néanmoins, il est préférable d'utiliser au possible la version stream puisque celle-ci est beaucoup plus rapide.



Lancement du traitement Mapping synchrone

Pour plus d'informations voir la page [mapSoapRunStreamFromPost](#)

Création d'un client

Le Web Services Description Language **wsdl** est un langage basé sur XML fournissant un modèle pour décrire les web services.

Le wsdl spécifie :

- les opérations possibles (le nom des méthodes)
- le détail des types de données échangées (nombre et types des paramètres en entrée et sortie de méthode)
- le protocole à utiliser : SOAP
- l'URL du service à consommer.

Coté client, un **stub** ainsi que des **interfaces** représentant les objets à manipuler sont générés automatiquement à l'import du wsdl.

Le wsdl est accessible en ajoutant ?wsdl à la fin de l'adresse où est présent le web service. Par exemple :

<http://localhost:8080/MapSoapServer/MapSoapServer?wsdl>

Installation du service web

Installation du service web :

- Vérifier que le fichier mapsoapserver est bien présent dans le dossier cgi de votre serveur web
- Tester la connection en local <http://localhost:8002/cgi-bin/mapsoapserver.exe>
- Puis en distant <http://localhost:8002/cgi-bin/mapsoapserver.exe?wsdl>

Liste des variables à vérifier et à configurer dans mapping.conf :

- SOAP_PATH_TEMP : répertoire temporaire utilisé par le server soap.
- SOAP_PATH_LOG : répertoire utilisé pour stocker la log uniquement si folder SOAP_DEBUG=on ou si une erreur apparaît.
- SOAP_LOCATION : URL distant d'accès au service web.
- SOAP_LOCAL_URL : répertoire local correspondant à SOAP_DOCUMENT_URL pour permettre au client SOAP de récupérer des fichiers volumineux.
- SOAP_DOCUMENT_URL : url distant d'accès au document produit par le serveur SOAP.
- SOAP_HOST_PROXY : proxy à utiliser pour accéder aux fichiers indiqués par le client SOAP.
- SOAP_PORT_PROXY : port du proxy.
- SOAP_DEBUG : le serveur SOAP est-il en mode debug.

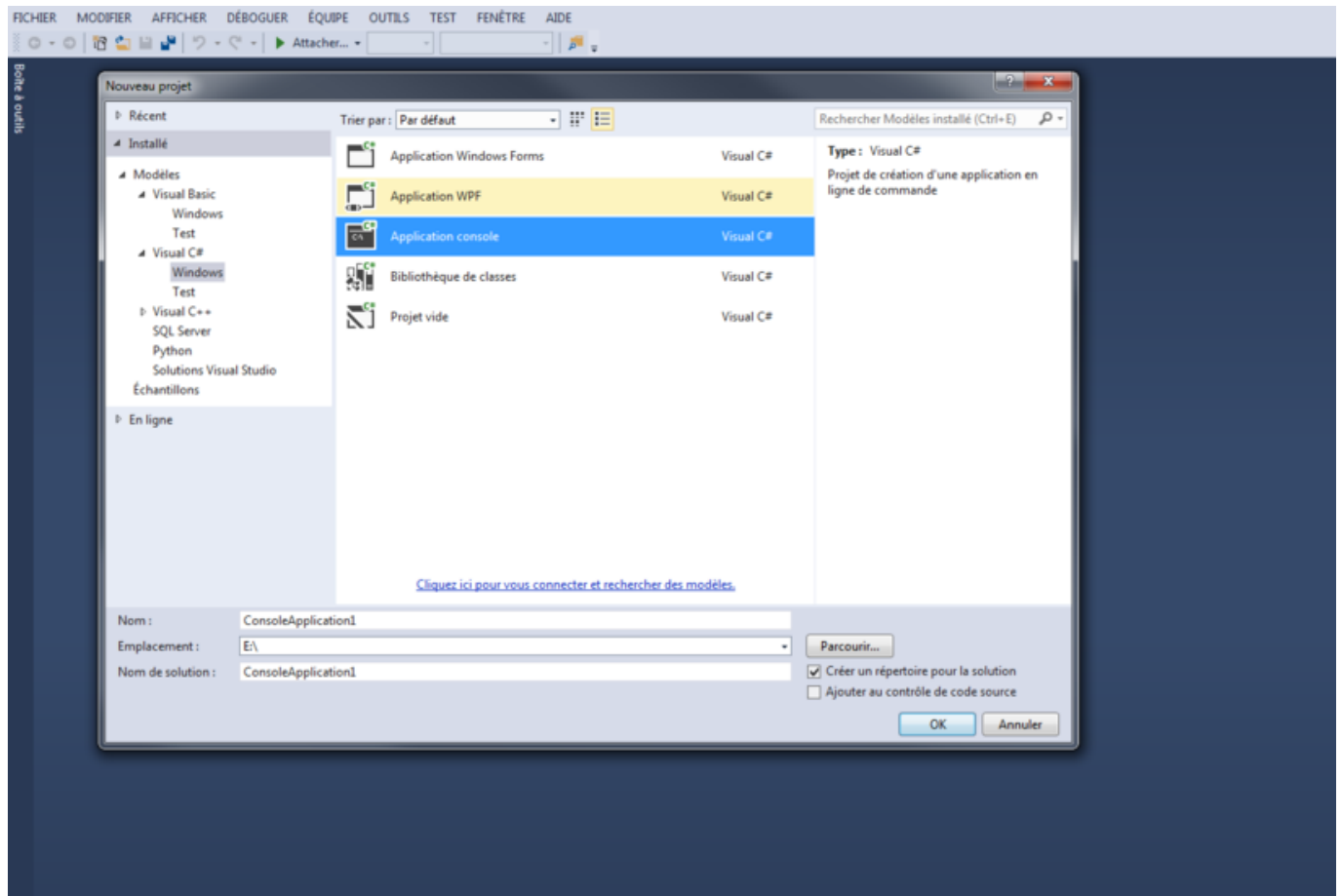
#SOAP

```
[SOAP_PATH_TEMP]          C:\Program Files\MappingWindows\Temp\HTTPServer
[SOAP_PATH_LOG]   C:\Program Files\MappingWindows\Temp\HTTPServer
[SOAP_LOCATION]   <a
href="http://localhost:8002/cgi-bin/mapsoapserver.exe">http://localhost:8002/
cgi-bin/mapsoapserver.exe</a>
[SOAP_LOCAL_URL]          C:\Program
Files\MappingWindows\MapHTTPServer\shareFolder
[SOAP_DOCUMENT_URL]       <a
href="http://localhost:8002/shareFolder">http://localhost:8002/shareFolder</a
>
[SOAP_HOST_PROXY]         0
[SOAP_PORT_PROXY]        0
[SOAP_DEBUG]      On
```

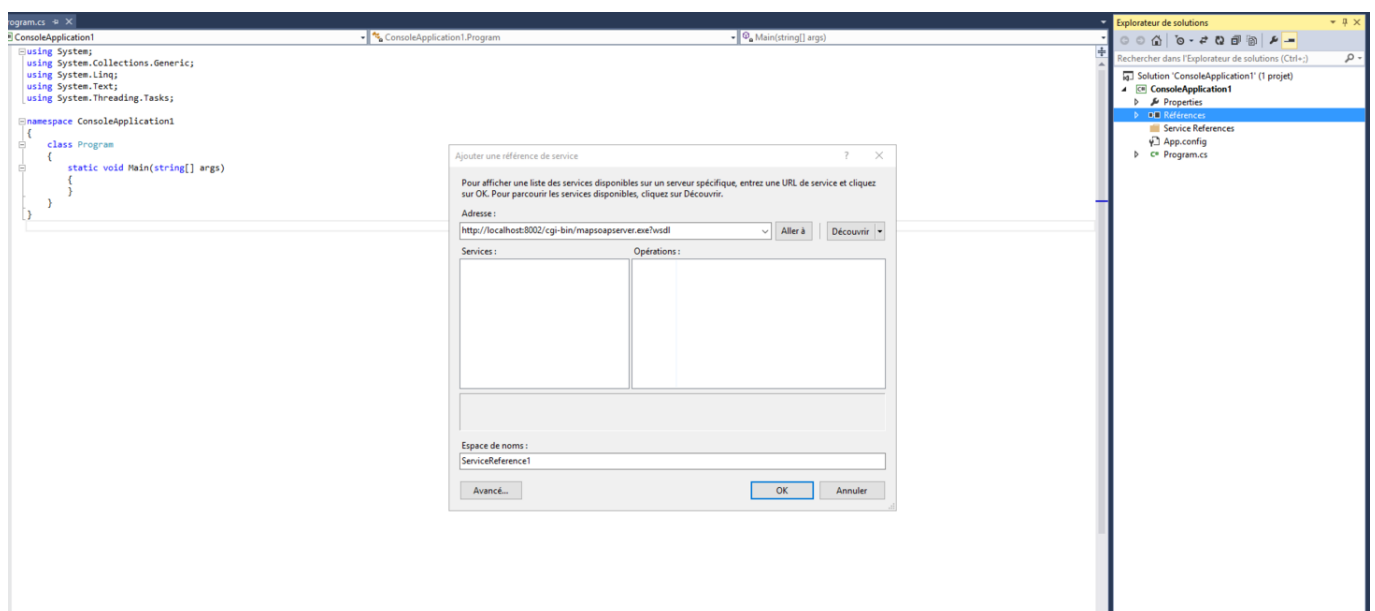
Exemple de source

Microsoft Visual Studio C#

Pour tester le web service mapping sous VS, faire un nouveau projet. Choisir le type de projet, par exemple ici une application console.



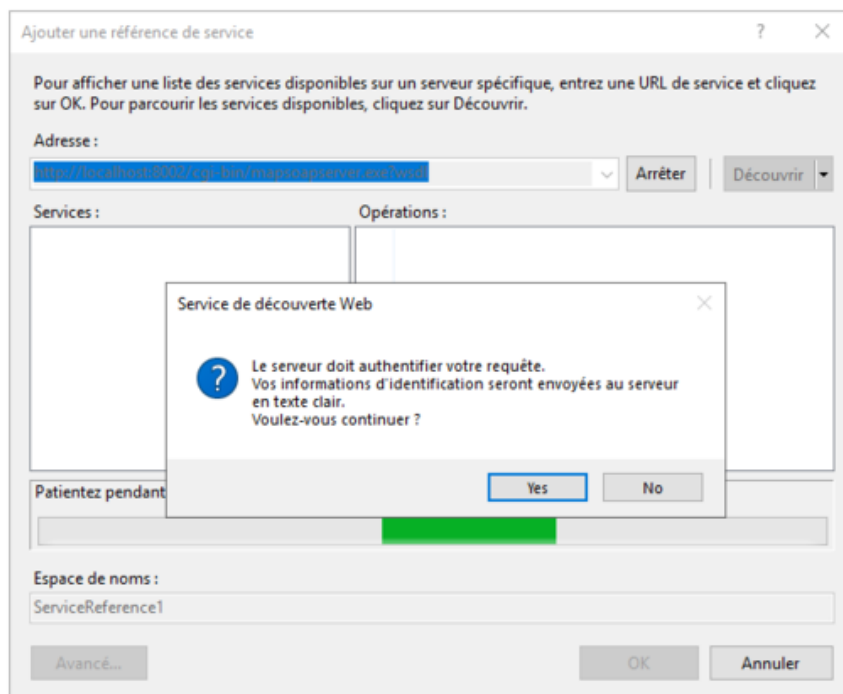
Ajouter un web référence



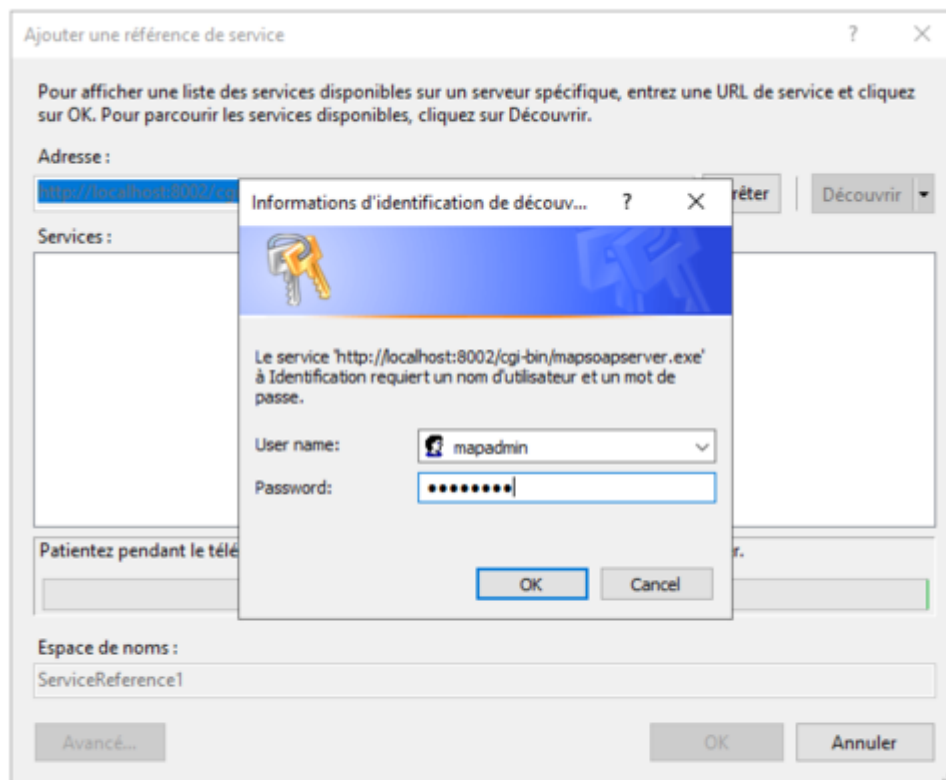
Comme adresse dans la référence web,

sélectionnera <http://localhost:8002/cgi-bin/mapsoapserver.exe?wsdl> et mettre le nom par exemple : mappingSuitePrintService

Répondre aux questions de sécurité, fonction du réglage du serveur web.



Identifiants : mapadmin/mapadmin



Taper enfin le code source pour tester.

```
using System;
using ConsoleApplication.mappingSuitePrintService;
namespace ConsoleApplication
{
    /// <summary>
    /// Summary description for Class1.
    /// </summary>
    class Class1
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main(string[] args)
        {
            MapSoapServerService mappingSoapServer = new
            MapSoapServerService();
            stopQueue
            resultStop=mappingSoapServer.mapStopQueue("PRT01");
        }
    }
}
```

Java

Création du stub

```
C:\jboss-4.2.1.GA\bin> wsconsume.bat -k -p clientmapsoapserver.jboss.ejb
<a href="http://127.0.0.1:8002/cgi-bin/mapsoapserver.exe?wsdl">http://127.0.0.1:8002/cgi-bin/mapsoapserver.exe?wsdl</a> -o
"C:\workspace\ClientMapSoapServer\src"
```

Fichier source java

```
package clientmapsoapserver.ConsoleApplication.sw;
public class ConsoleApplication
{
    public static void main(String[] args)
    {
        clientmapsoapserver.jboss.ejb.MapSoapServerService service = new
clientmapsoapserver.jboss.ejb.MapSoapServerService();
        clientmapsoapserver.jboss.ejb.MapSoapServer mappingSoapServer =
service.getMapSoapServerPort();
        mappingSoapServer.mapStopQueue("PRT01");
    }
}
```

Javascript

```
<html>
<body>
<script type="text/javascript">
    var http_request = false;
    function makeRequest(url) {
        var xhr_object = null;
        if ( window.XMLHttpRequest ) xhr_object = new XMLHttpRequest();
        else if(window.ActiveXObject) {
            try {
                xhr_object = new ActiveXObject("Msxml2.XMLHTTP");
            } catch(e) {
                try {
                    xhr_object = new ActiveXObject("Microsoft.XMLHTTP");
                } catch(e) {
                    xhr_object = null;
                }
            }
        }

        if (!xhr_object) {
            alert('Cannot create XMLHTTP instance');
            return false;
        }
    }
}
```

```

    }
    xhr_object.open('GET', url, false);
    xhr_object.send(parameter);
    return xhr_object.responseXML;
}

function stopQueue(qname) {
    currentresults="";
    parameter="<a
href="http://127.0.0.1:8002/cgi-bin/mapsoapserver.exe/mapStopQueue?qname=">ht
tp://127.0.0.1:8002/cgi-bin/mapsoapserver.exe/mapStopQueue?qname=</a>" + qname;
    var xmldoc=makeRequest(parameter);
    var root=xmldoc.getElementsByTagName('stopQueue').item(0);
    var results=root.getElementsByTagName('infos').item(0);
    if (results.textContent) {
        currentresults=results.textContent;
    }
    else if (results.text) {
        currentresults=results.text;
    }
    alert(currentresults);
}
stopQueue("MAPPING");
</script>
</body>
</html>

```

Autoriser les connections anonymes

Faire le fichier de configuration d'Apache. Pour permettre les connections anonymes, sur une autre URL ou un autre port, il suffit de définir un nouveau AccessFileName.

```

#BEGIN_MAPPING_1.00.30_12-6-2007
#LoadModule ldap_module modules/mod_ldap.so
#LoadModule authnz_ldap_module modules/mod_authnz_ldap.so
Listen 8002
NameVirtualHost *:8002
<VirtualHost *:8002>
    AccessFileName .htaccess
    ServerName SERVEURSAP
    DocumentRoot "C:/Program Files/MappingWindows/MapHTTPServer"
    <Directory "C:/Program Files/MappingWindows/MapHTTPServer/cgi-bin">
        AllowOverride All
        Order allow,deny
        Allow from all
        Options None
    </Directory>
    ScriptAlias /cgi-bin/ "C:/Program Files/MappingWindows/MapHTTPServer/cgi-
bin/"

```



```

<Directory "C:/Program Files/MappingWindows/MapHTTPServer">
    AllowOverride All
    Order allow,deny
    Allow from all
    Options None
</Directory>
</VirtualHost>
#END_MAPPING_1.00.30_12-6-2007
<Directory "C:\Program Files\Apache Software
Foundation\Apache2.2\htdocs\MapWebFiles">
    AllowOverride all
    Order deny,allow
    </Directory>

AccessFileName .config
ScriptAlias /MapHTTPServer "C:/Program Files/MappingWindows/MapHTTPServer/"
<Directory "C:/Program Files/MappingWindows/MapHTTPServer">
    AllowOverride All
    Order allow,deny
    Allow from all
    Options None
</Directory>
ScriptAlias /MapHTTPServer/cgi-bin "C:/Program
Files/MappingWindows/MapHTTPServer/cgi-bin/"
<Directory "C:/Program Files/MappingWindows/MapHTTPServer/cgi-bin">
    AllowOverride All
    Order allow,deny
    Allow from all
    Options None
</Directory>

```

Liste des fonctions existantes

Pour découvrir les fonctions disponibles utiliser
le **wsdl** <http://127.0.0.1:8002/cgi-bin/mapsoapserver.exe?wsdl>

sinon avec l'accès à la machine en localhost la présentation HTML des
fonctions <http://127.0.0.1:8002/cgi-bin/mapsoapserver.exe>

Remarque :

Sur Linux supprimer le .exe.

L'ensemble des fonctions sont décrites dans la page HTML de présentation des fonctions. Le WSDL décrit les paramètres attendus en entrée et en sortie. Les paramètres particuliers key et value vont par paire et permettent de définir un ensemble infini de valeurs pour pouvoir les utiliser dans le moteur de règles. Par exemple, dans le cas des commandes d'insertion de spool, cela devient des users datas des fichiers spools produit. Dans le cas des commandes de moteur de règles, cela devient des paramètres du moteur de règles.

Fonctions de listage

mapListSplf : List user spool file

Sample GET

```
<a  
href="http://localhost:8002/cgi-bin/mapsoapserver.exe/mapListSplf?login=mapad  
min&ownerName=*&site=*&qname=*&splf=*&state=*>http://loc  
alhost:8002/cgi-  
bin/mapsoapserver.exe/mapListSplf?login=mapadmin&ownerName=*&site=*&  
mp;qname=*&splf=*&state=*</a>
```

mapListSplfWithDate : List user spool file by date

Sample GET

```
<a  
href="http://localhost:8002/cgi-bin/mapsoapserver.exe/mapListSplfWithDate?log  
in=mapadmin&ownerName=*&site=*&qname=*&splf=*&state=*&  
;">http://localhost:8002/cgi-bin/mapsoapserver.exe/mapListSplfWithDate?login=  
mapadmin&ownerName=*&site=*&qname=*&splf=*&state=*&  
<a>  
fromDay=21&fromMonth=1&fromYear=2006&toDay=21&toMonth=1&t  
oYear=2006
```

mapListPrinter : List available printer

Sample GET

```
<a  
href="http://localhost:8002/cgi-bin/mapsoapserver.exe/mapListPrinter?login=ma  
padmin&ownerName=*&site=*&qname=*&type=*&state=*>http://  
localhost:8002/cgi-  
bin/mapsoapserver.exe/mapListPrinter?login=mapadmin&ownerName=*&site=  
*&qname=*&type=*&state=*</a>
```

mapListPrinterTransfer : List all printer available for transfer

Sample GET

```
<a  
href="http://localhost:8002/cgi-bin/mapsoapserver.exe/mapListPrinterTransfer?  
login=mapadmin">http://localhost:8002/cgi-bin/mapsoapserver.exe/mapListPrinte  
rTransfer?login=mapadmin</a>
```

Fonctions sur un fichier spool

- mapRestartPrinting : Restart a print at a given page

Sample GET

```
<a
href="http://localhost:8002/cgi-bin/mapsoapserver.exe/mapRestartPrinting?jobNum=000000000030072&startPage=1&stopPage=1&";>http://localhost:8002/cgi-bin/mapsoapserver.exe/mapRestartPrinting?jobNum=000000000030072&startPage=1&stopPage=1&;</a>
user=mapadmin&
qname=PRT01&;title=My Title&;originalQname=PRT01
```

- mapMoveSplf : Move a spool file to another queue

Sample GET

```
<a
href="http://localhost:8002/cgi-bin/mapsoapserver.exe/mapMoveSplf?jobNum=000000000030072&qname=PRT01&oldqname=PRT02">http://localhost:8002/cgi-bin/mapsoapserver.exe/mapMoveSplf?jobNum=000000000030072&qname=PRT01&oldqname=PRT02</a>
```

- mapDelSplf : Delete a spool file

Sample GET

```
<a
href="http://localhost:8002/cgi-bin/mapsoapserver.exe/mapDelSplf?jobNum=000000000030072&qname=PRT01">http://localhost:8002/cgi-bin/mapsoapserver.exe/mapDelSplf?jobNum=000000000030072&qname=PRT01</a>
```

- mapReleaseSplf : Release a spool file

Sample GET

```
<a
href="http://localhost:8002/cgi-bin/mapsoapserver.exe/mapReleaseSplf?jobNum=000000000030072&qname=PRT01">http://localhost:8002/cgi-bin/mapsoapserver.exe/mapReleaseSplf?jobNum=000000000030072&qname=PRT01</a>
```

- mapHoldSplf : Hold a spool file

Sample GET

```
<a
href="http://localhost:8002/cgi-bin/mapsoapserver.exe/mapHoldSplf?jobNum=000000000030072&qname=PRT01">http://localhost:8002/cgi-bin/mapsoapserver.exe/mapHoldSplf?jobNum=000000000030072&qname=PRT01</a>
```

- mapInfoSplf : Display info of spool file

Sample GET

```
<a
href="http://localhost:8002/cgi-bin/mapsoapserver.exe/mapInfoSplf?jobNum=000000000030072&qname=PRT01&user=mapadmin">http://localhost:8002/cgi-bin/
```

mapsoapserver.exe/mapInfoSplf?jobNum=000000000030072&qname=PRT01&user=mapadmin

- mapChangeSplfUserData : Change User Data of a spool file

Sample GET

```
<a
href="http://localhost:8002/cgi-bin/mapsoapserver.exe/mapChangeSplfUserData?jobNum=000000000030072&qname=PRT01&key=MYUSERDATA&value=my">http://localhost:8002/cgi-bin/mapsoapserver.exe/mapChangeSplfUserData?jobNum=000000000030072&qname=PRT01&key=MYUSERDATA&value=my</a>
value
```

- mapChangeSplfGroupID : Change Group ID of a spool file

Sample GET

```
<a
href="http://localhost:8002/cgi-bin/mapsoapserver.exe/mapChangeSplfGroupID?jobNum=000000000030072&qname=PRT01&value=my">http://localhost:8002/cgi-bin/mapsoapserver.exe/mapChangeSplfGroupID?jobNum=000000000030072&qname=PRT01&value=my</a>
value
```

- mapChangeSplfAccountancyCode : Change Accountancy Code of a spool file

Sample GET

```
<a
href="http://localhost:8002/cgi-bin/mapsoapserver.exe/mapChangeSplfAccountancyCode?jobNum=000000000030072&qname=PRT01&value=0">http://localhost:8002/cgi-bin/mapsoapserver.exe/mapChangeSplfAccountancyCode?jobNum=000000000030072&qname=PRT01&value=0</a>
```

- mapChangeSplfStateAttribute : Change State Attribute of a spool file

Sample GET

```
<a
href="http://localhost:8002/cgi-bin/mapsoapserver.exe/mapChangeSplfStateAttribute?jobNum=000000000030072&qname=PRT01&value=my">http://localhost:8002/cgi-bin/mapsoapserver.exe/mapChangeSplfStateAttribute?jobNum=000000000030072&qname=PRT01&value=my</a>
value
```

- mapChangeSplfRetentionTime : Change Retention Time of a spool file

Sample GET

```
<a
href="http://localhost:8002/cgi-bin/mapsoapserver.exe/mapChangeSplfRetentionT
```

ime?jobNum=000000000030072&qname=PRT01&value=my">http://localhost:8002/cgi-bin/mapsoapserver.exe/mapChangeSplfRetentionTime?jobNum=000000000030072&qname=PRT01&value=my value

- mapChangeSplfPriority : Change Priority of a spool file

Sample GET

http://localhost:8002/cgi-bin/mapsoapserver.exe/mapChangeSplfPriority?jobNum=000000000030072&qname=PRT01&value=my value

- mapChangeSplfStopPage : Change Stop Page of a spool file

Sample GET

http://localhost:8002/cgi-bin/mapsoapserver.exe/mapChangeSplfStopPage?jobNum=000000000030072&qname=PRT01&value=my value

- mapChangeSplfStartPage : Change Start Page of a spool file

Sample GET

http://localhost:8002/cgi-bin/mapsoapserver.exe/mapChangeSplfStartPage?jobNum=000000000030072&qname=PRT01&value=my value

- mapChangeSplfNbCopy : Change Number of Copy of a spool file

Sample GET

http://localhost:8002/cgi-bin/mapsoapserver.exe/mapChangeSplfNbCopy?jobNum=000000000030072&qname=PRT01&value=my value

- mapChangeSplfTitle : Change Title of a spool file

Sample GET

http://localhost:8002/cgi-bin/mapsoapserver.exe/mapChangeSplfTitle?jobNum=000000000030072&qname=PRT01&value=my value

- mapChangeSplfUser : Change User of a spool file

Sample GET

http://localhost:8002/cgi-bin/mapsoapserver.exe/mapChangeSplfUser?jobNum=000000000030072&qname=PRT01&value=my value

- mapChangeSplfSave : Change Save of a spool file

Sample GET

http://localhost:8002/cgi-bin/mapsoapserver.exe/mapChangeSplfSave?jobNum=000000000030072&qname=PRT01&value=my value

- mapLogSplf : View log of the spool file

Sample GET

http://localhost:8002/cgi-bin/mapsoapserver.exe/mapLogSplf?qname=PRT01&jobNum=000000000030072&login=yourvalue&filter=yourvalue dateFrom=yourvalue&dateTo=yourvalue&level=yourvalue

Fonctions sur plusieurs fichiers spool

- mapHoldMultiSplf : Hold multi spool file

Sample GET

http://localhost:8002/cgi-bin/mapsoapserver.exe/mapHoldMultiSplf?jobNum=000000000030072&qname=PRT01

- mapReleaseMultiSplf : Release multi spool file

Sample GET

http://localhost:8002/cgi-bin/mapsoapserver.exe/mapReleaseMultiSplf?jobNum=000000000030072&qname=PRT01

- mapDelMultiSplf : Delete multi spool file

Sample GET

http://localhost:8002/cgi-bin/mapsoapserver.exe/mapDelMultiSplf?jobNum=000000000030072&qname=PRT01

- mapMoveMultiSplf : Move multi spool file

Sample GET

http://localhost:8002/cgi-bin/mapsoapserver.exe/mapMoveMultiSplf?jobNum=000000000030072&qname=PRT01&qname=oldqname=PRT02

- mapSplfPreviewURL : Download the url of the preview spool file

Sample GET

http://localhost:8002/cgi-bin/mapsoapserver.exe/mapSplfPreviewURL?qname=PRT01&qname=jobNum=000000000030072

- mapSplfPreviewStream : Download the stream of the preview spool file

Sample GET

http://localhost:8002/cgi-bin/mapsoapserver.exe/mapSplfPreviewStream?qname=PRT01&qname=jobNum=000000000030072

Fonctions sur les imprimantes

- mapRefreshStatus : Refresh printer status

Sample GET

http://localhost:8002/cgi-bin/mapsoapserver.exe/mapRefreshStatus?qname=PRT01

- mapRestartDevice : Restart a device

Sample GET

```
<a
href="http://localhost:8002/cgi-bin/mapsoapserver.exe/mapRestartDevice?device
Name=DEVPRT01&qname=PRT01">http://localhost:8002/cgi-bin/mapsoapserver.ex
e/mapRestartDevice?deviceName=DEVPRT01&qname=PRT01</a>
```

- mapStartQueue : Start a queue

Sample GET

```
<a
href="http://localhost:8002/cgi-bin/mapsoapserver.exe/mapStartQueue?qname=PRT
01">http://localhost:8002/cgi-bin/mapsoapserver.exe/mapStartQueue?qname=PRT01
</a>
```

- mapStopQueue : Stop a queue

Sample GET

```
<a
href="http://localhost:8002/cgi-bin/mapsoapserver.exe/mapStopQueue?qname=PRT0
1">http://localhost:8002/cgi-bin/mapsoapserver.exe/mapStopQueue?qname=PRT01</
a>
```

- mapInfoPrinter : Display info of printer

Sample GET

```
<a
href="http://localhost:8002/cgi-bin/mapsoapserver.exe/mapInfoPrinter?qname=PR
T01&user=mapadmin">http://localhost:8002/cgi-bin/mapsoapserver.exe/mapInf
oPrinter?qname=PRT01&user=mapadmin</a>
```

- mapLogQueue : View log of a queue

Sample GET

```
<a
href="http://localhost:8002/cgi-bin/mapsoapserver.exe/mapLogQueue?qname=PRT01
&login=mapadmin&filter=yourvalue&dateFrom=yourvalue&";>http://
localhost:8002/cgi-
bin/mapsoapserver.exe/mapLogQueue?qname=PRT01&login=mapadmin&filter=y
ourvalue&dateFrom=yourvalue&
dateTo=yourvalue&level=yourvalue
```

Fonctions sur les demandes d'impression

- mapUploadMessage : Upload a message as a spool file

Cette fonction permet d'insérer un message dans le spooler en envoyant le message en texte en tant que paramètre.

- La method permet par exemple de déclencher un traitement.
- La method ne convient pas pour les données binaires ou de taille importante.

Sample GET

```
<a
href="http://localhost:8002/cgi-bin/mapsoapserver.exe/mapUploadMessage?qname=
PRT01&title=yourvalue&user=yourvalue&stream=yourvalue&";>http:
//localhost:8002/cgi-
bin/mapsoapserver.exe/mapUploadMessage?qname=PRT01&title=yourvalue&us
er=yourvalue&stream=yourvalue&;</a>
key=yourvalue&;value=yourvalue
```

- mapUploadFile : Upload a file as a spool file

Cette fonction permet d'insérer un fichier dans le spooler en envoyant le fichier en tant que stream(encodée en base64)

La method permet de transférer un fichier binaires mais ne permet pas de supporter les tailles importantes.

Sample GET

```
<a
href="http://localhost:8002/cgi-bin/mapsoapserver.exe/mapUploadFile?qname=PRT
01&title=yourvalue&user=yourvalue&stream=yourvalue&";>http://l
ocalhost:8002/cgi-
bin/mapsoapserver.exe/mapUploadFile?qname=PRT01&title=yourvalue&user=
yourvalue&stream=yourvalue&;</a>
key=yourvalue&;value=yourvalue
```

- mapUploadURL : Upload a url as a spool file

Cette fonction permet d'insérer un fichier dans le spooler en demandant au service web mapping d'aller chercher le fichier sur une url donnée en paramètre.Cette fonction permet de transférer des fichiers binaires et de taille importante.

Sample GET

```
<a
href="http://localhost:8002/cgi-bin/mapsoapserver.exe/mapUploadURL?qname=PRT0
1&title=yourvalue&user=yourvalue&url=yourvalue&";>http://local
host:8002/cgi-
bin/mapsoapserver.exe/mapUploadURL?qname=PRT01&title=yourvalue&user=y
ourvalue&url=yourvalue&;</a>
key=yourvalue&;value=yourvalue
```

- mapRunStream : Run the stream of the mapping engine

Cette fonction soumet au moteur de règles un fichier encodé en base 64 et reçoit en retour un document encodé en base 64.

Le moteur de règles a alors la valeur MAP_SOAP = On activé automatiquement permettant de faire des règles particulières sur les accès SOAP.

Sample GET

```
<a  
href="http://localhost:8002/cgi-bin/mapsoapserver.exe/mapRunStream?stream=base64 data&key=anyname&value=any value  
>
```

Remarque :

Pour le cas particulier de l'accès SOAP avec une requête de type GET, si vous ajoutez le paramètre « getObject=1 » vous recevez en réponse directement le document en retour sans encodage base 64. Ce qui permet de l'utiliser dans des pages HTML pour demander un preview d'un document par exemple.

<http://localhost:8002/cgi-bin/mapsoapserver.exe/mapRunStream?stream=base64 data&key=anyname&value=any value&getObject=1>

- mapRunURL : Run the stream and get URL

Cette fonction soumet au moteur de règles un fichier encodé en base 64 et reçoit en retour l'URL du document produit, charge au client web de supprimer le document par la requête mapRemoveTempURL ou tout script de son choix.

Le moteur de règles a alors la valeur MAP_SOAP = On activé automatiquement permettant de faire des règles particulières sur les accès SOAP.

Sample GET

```
<a  
href="http://localhost:8002/cgi-bin/mapsoapserver.exe/mapRunURL?stream=yourvalue&key=yourvalue&value=yourvalue">http://localhost:8002/cgi-bin/mapsoapserver.exe/mapRunURL?stream=yourvalue&key=yourvalue&value=yourvalue</a>
```

- mapRemoveTempURL : Remove the temp URL built by SOAP server

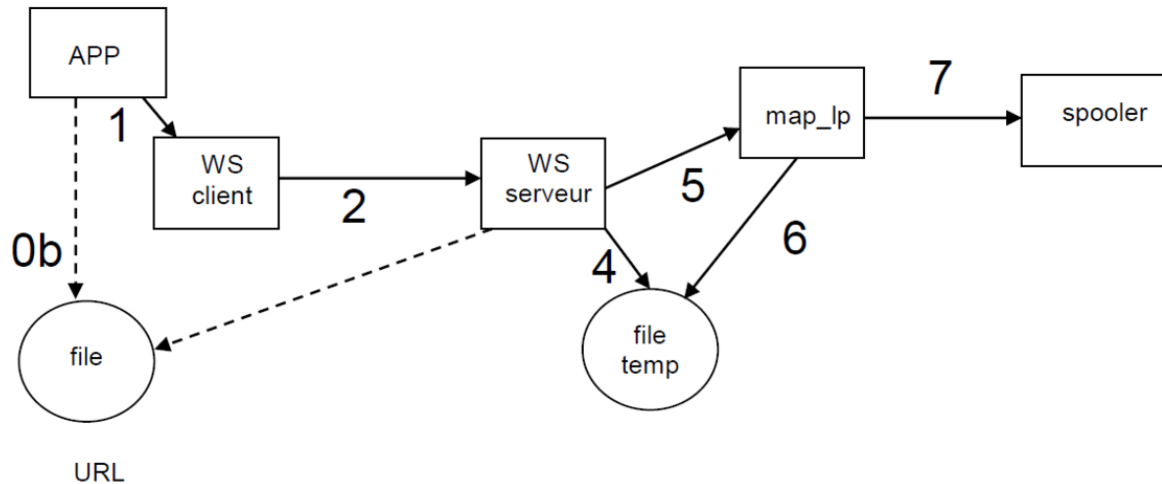
Cette fonction permet de supprimer le fichier créé par la méthode mapRunURL ou mapSplfPreviewURL.

Sample GET

```
<a  
href="http://localhost:8002/cgi-bin/mapsoapserver.exe/mapRemoveTempURL?downloadurl=http://localhost/shareFolder/0904324242343243.tmp">http://localhost:8002/cgi-bin/mapsoapserver.exe/mapRemoveTempURL?downloadurl=http://localhost/shareFolder/0904324242343243.tmp</a>
```

Détails sur la méthode upload

La méthode upload permet d'envoyer un spool dans une file d'attente. Un fichier temporaire sera créé côté serveur contenant les données envoyées. Ce fichier sera bien entendu supprimé une fois la commande map_lp exécutée. La version URL nécessite au préalable d'avoir déposé son fichier sur un site accessible au service web. Ci dessous, le schéma présentant l'architecture utilisée lors d'un upload.



Légende

0b Dépôt du fichier sur une URL distante (version URL)

1 Lancement du client

2 Appel de la méthode à distance

3b Lecture du fichier à l'URL indiquée (version URL)

4 Création du fichier temporaire

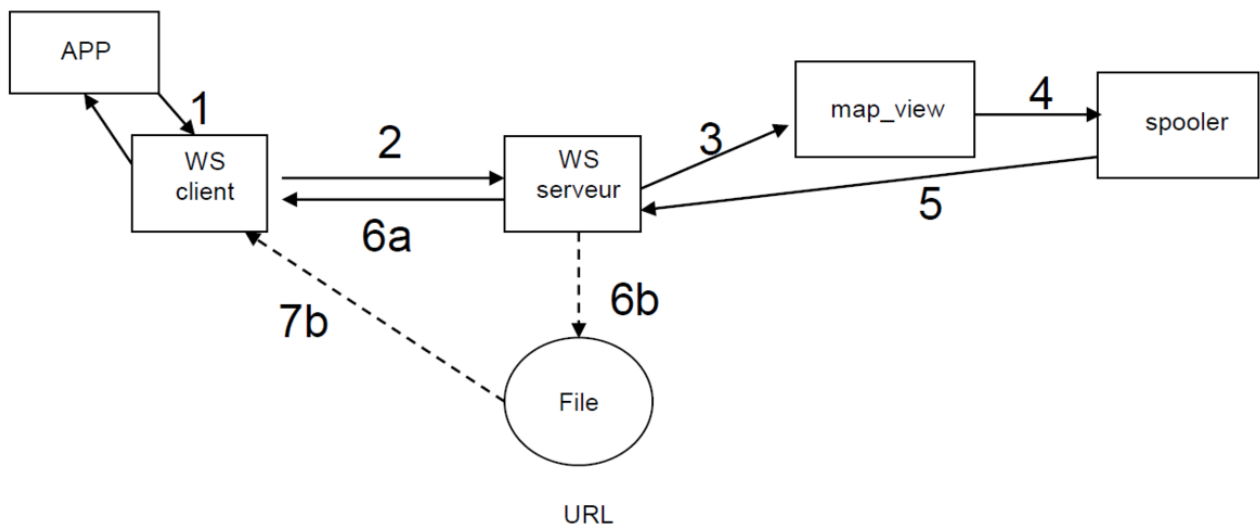
5 Lancement de map_lp

6 Lecture des données du fichier temporaire

7 Création du spool dans la file d'attente

Détails sur la méthode preview

La méthode preview consiste à renvoyer le contenu d'un spool qui a été préalablement envoyé dans une file d'attente. Ci dessous, le schéma présentant l'architecture utilisée lors d'une preview.



Légende

- 1 Lancement du client
- 2 Appel de la méthode à distance
- 3 Lancement de map_view
- 4 Communication avec le spooler
- 5 Réception des résultats par le web service
- 6a Envoi des données en stream au client
- 6b Ecriture du résultat dans un fichier
- 7a Retourne le spool + headers au client
- 7b Envoie l'URL au client
- 8b Envoie l'URL à l'application

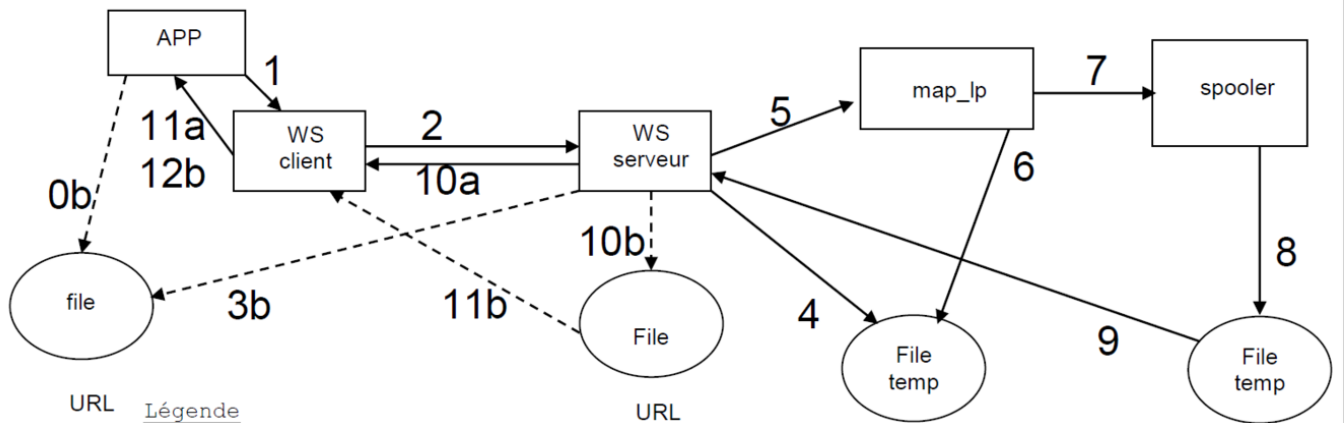
Détails sur la méthode run

La méthode run permet de remaquetter le spool envoyé. Le run peut se voir comme 3 étapes successives : un upload, un traitement puis une prévisualisation du fichier généré. La différence par rapport à un upload classique est que dans le cas du run, le spool n'a pas à être envoyé dans une queue pour être imprimé mais dans une entry.

Ainsi, le spool passe par le moteur de règles. Un user data nommé SOAP_RUN_FILE est ajouté automatiquement.

Une règle testant la présence de ce user data permettra donc d'y faire un traitement spécifique : appel de mapcpysplf puis avec les différents arguments (mapnam, mapseq, mapmod, inFile, outFile et lang) permettant ce remaquettage. Ci dessous, le schéma présentant l'architecture utilisée lors

d'un run.



0b Dépôt du fichier sur une URL distante (version URL)

1 Lancement du client

2 Appel de la méthode à distance

3b Lecture du fichier à l'URL indiquée

4 Création du fichier temporaire

5 Lancement de map_lp

6 Lecture des données du fichier temporaire

7 Création du spool dans la file d'attente

8 Traitement et création du fichier temporaire (fichier remaquetté)

9 Lecture du fichier par le web service

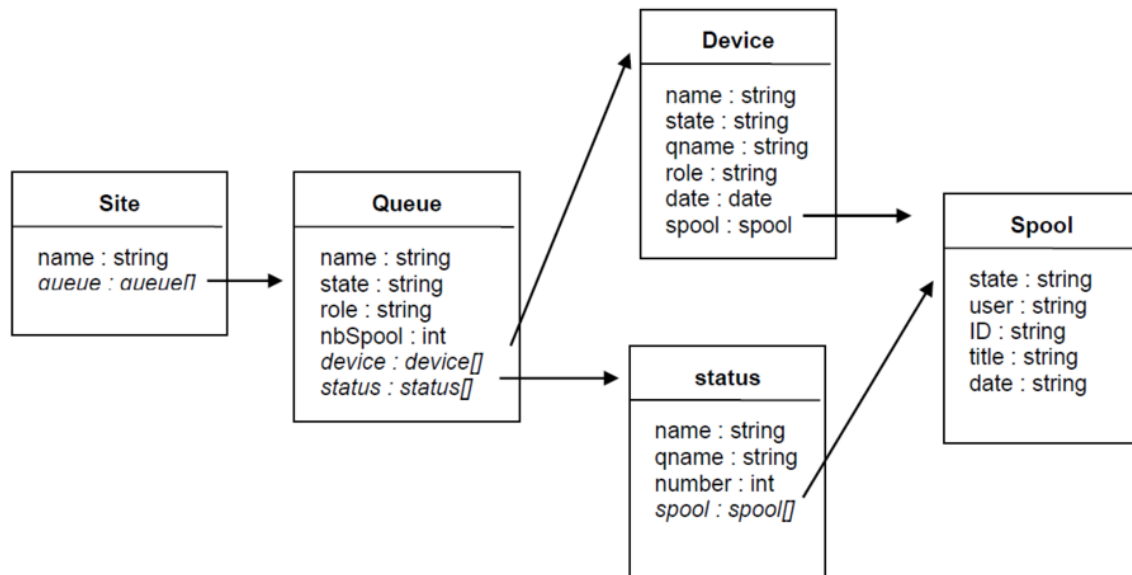
10a Envoi des données en stream au client

10b Ecriture du résultat dans un fichier

11a Retourne le contenu du spool remaquetté au client

11b Retourne l'URL du spool remaquetté au client

Schéma de données XML pour la vue des spools



Comment renvoyer un fichier au client du service web depuis le moteur de règles

Afin que le moteur de règles puisse renvoyer des réponses au service web, vous devez utiliser le moteur de règles.

```
"map_echo.exe" "Cache-Control: no-cache\nContent-type: application/pdf;\n\n"
"map_echo.exe" "-infile:C:Program FilesMappingWindowsTempmyfile.pdf"
```

Protocole utilisé

Le protocole utilisé est décrit dans <http://www.w3.org/TR/soap>

SOAP HTTP GET Usage

Question :

```
GET /travelcompany.example.org/reservations?code=FT35ZBQ HTTP/1.1
Host: travelcompany.example.org
Accept: text/html;q=0.5, application/soap+xml
```

Réponse :

```
HTTP/1.1 200 OK
Content-Type: application/xml; charset="utf-8"
Content-Length: nnnn
<?xml:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  >http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:x="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  >
```

```

href="http://travelcompany.example.org/vocab#">http://travelcompany.example.o
rg/vocab#</a>"
  env:encodingStyle="<a
href="http://www.w3.org/1999/02/22-rdf-syntax-ns#">http://www.w3.org/1999/02/
22-rdf-syntax-ns#</a>"&gt;
  <lt;x:ReservationRequest
rdf:about="<a
href="http://travelcompany.example.org/reservations?code=FT35ZBQ">http://trav
elcompany.example.org/reservations?code=FT35ZBQ</a>"&gt;
    <lt;x:passenger&gt;Åke Jógvan Øyvind&lt;/x:passenger&gt;
    <lt;x:outbound&gt;
    <lt;x:TravelRequest&gt;
      <lt;x:to&gt;LAX&lt;/x:to&gt;
      <lt;x:from&gt;LGA&lt;/x:from&gt;
      <lt;x:date&gt;2001-12-14&lt;/x:date&gt;
    <lt;/x:TravelRequest&gt;
    <lt;/x:outbound&gt;
    <lt;x:return&gt;
    <lt;x:TravelRequest&gt;
      <lt;x:to&gt;JFK&lt;/x:to&gt;
      <lt;x:from&gt;LAX&lt;/x:from&gt;
      <lt;x:date&gt;2001-12-20&lt;/x:date&gt;
    <lt;/x:TravelRequest&gt;
    <lt;/x:return&gt;
  <lt;/x:ReservationRequest&gt;

```

SOAP HTTP POST Usage

Question :

```

POST /Reservations HTTP/1.1
Host: travelcompany.example.org
Content-Type: application/soap+xml; charset="utf-8"
Content-Length: nnnn
&lt;?xml version='1.0'&nbsp;?&gt;
&lt;env:Envelope xmlns:env="<a
href="http://www.w3.org/2003/05/soap-envelope">http://www.w3.org/2003/05/soap
-envelope</a>" &gt;
  <lt;env:Header&gt;
    <lt;t:transaction
      xmlns:t="<a
href="http://thirdparty.example.org/transaction">http://thirdparty.example.or
g/transaction</a>"
      env:encodingStyle="<a
href="http://example.com/encoding">http://example.com/encoding</a>"
      env:mustUnderstand="true" &gt;5&lt;/t:transaction&gt;
    <lt;/env:Header&gt;
    <lt;env:Body&gt;
      <lt;m:chargeReservation
        env:encodingStyle="<a

```

```

href="http://www.w3.org/2003/05/soap-encoding">http://www.w3.org/2003/05/soap
-encoding</a>"
    xmlns:m="<a
href="http://travelcompany.example.org/">http://travelcompany.example.org/</a
>"&gt;
    <lt;m:reservation xmlns:m="<a
href="http://travelcompany.example.org/reservation">http://travelcompany.exam
ple.org/reservation</a>"&gt;
        <lt;m:code&gt;FT35ZBQ&lt;/m:code&gt;
        <lt;/m:reservation&gt;
        <lt;o:creditCard xmlns:o="<a
href="http://mycompany.example.com/financial">http://mycompany.example.com/fi
nancial</a>"&gt;
            <lt;n:name xmlns:n="<a
href="http://mycompany.example.com/employees">http://mycompany.example.com/em
ployees</a>"&gt;
                Åke Jógvan Øyvind
            <lt;/n:name&gt;
            <lt;o:number&gt;1234567890999999&lt;/o:number&gt;
            <lt;o:expiration&gt;2005-02&lt;/o:expiration&gt;
            <lt;/o:creditCard&gt;
            <lt;/m:chargeReservation
        <lt;/env:Body&gt;
    <lt;/env:Envelope&gt;

```

Réponse :

```

HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset="utf-8"
Content-Length: nnnn
&lt;?xml version='1.0'&nbsp;&?&gt;
&lt;env:Envelope xmlns:env="<a
href="http://www.w3.org/2003/05/soap-envelope">http://www.w3.org/2003/05/soap
-envelope</a>" &gt;
    <lt;env:Header&gt;
        ...
        ...
    <lt;/env:Header&gt;
    <lt;env:Body&gt;
        ...
        ...
    <lt;/env:Body&gt;
&lt;/env:Envelope&gt;

```