

Université IBM i

10 et 11 mai 2016 – IBM Client Center de Bois-Colombes

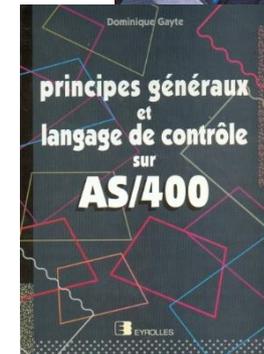
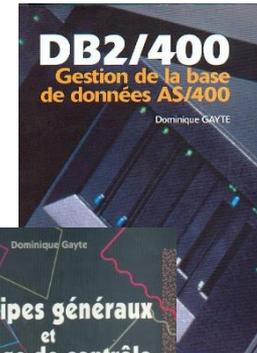
S45 - Les différentes techniques d'intégration PHP + RPG

Mercredi 11 mai – 15h15-16h45

Gautier DUMAS – NoToS
Dominique GAYTE - NoToS

NoToS

- Expertise autour de l'IBM i
 - Sécurité
 - Regard moderne (DB2 Web Query)
- Service
 - Formation, audit, développement...
- PHP sur IBM i avec Zend
- Développement de progiciels
 - Modernisation à valeur ajoutée des IBM i



Valorisation des spools des IBM i (AS/400)
Transformation en PDF, archivage, indexation
<http://www.notos.fr/phpSpool.aspx>



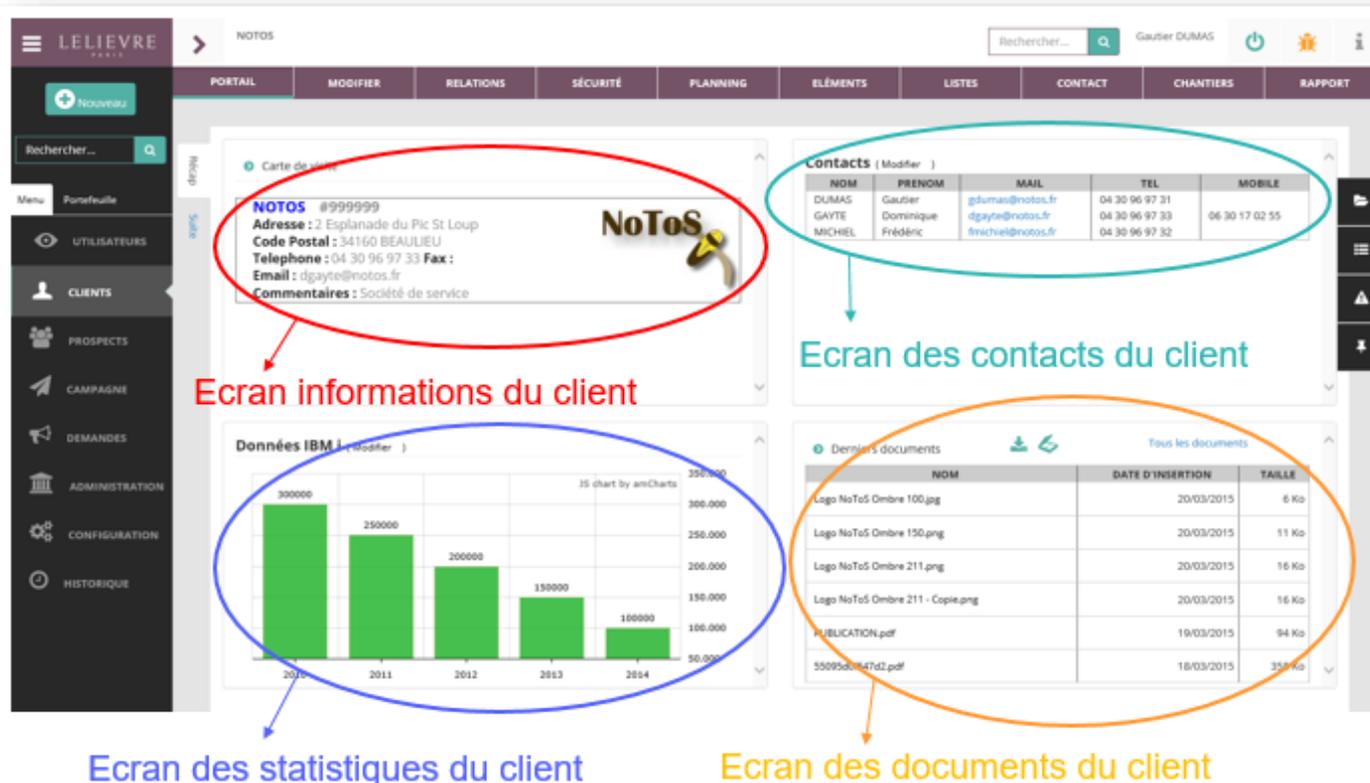
Gestion de Contenu (ECM)
GED, graphiques, alertes, workflow, GANTT...
<http://www.lorena.pro>

Sommaire

- Pourquoi se tourner vers des solutions PHP+RPG
- Appeler du PHP depuis RPG
- Appeler du RPG depuis PHP
- Exposer une logique RPG par web service grâce à PHP

Pourquoi se tourner vers des solutions PHP+RPG

- Apporter les applications RPG existantes dans les navigateurs web et les smartphones rapidement et simplement sans réécriture à 100 % de l'application



NOTOS #999999
 Adresse : 2 Esplanade du Pic St Loup
 Code Postal : 34160 BEAULIEU
 Telephone : 04 30 96 97 33 Fax :
 Email : dgayte@notos.fr
 Commentaires : Société de service

NOM	PRENOM	MAIL	TEL	MOBILE
DUMAS	Gautier	gdumas@notos.fr	04 30 96 97 31	
GAYTE	Dominique	dgayte@notos.fr	04 30 96 97 33	06 30 17 02 55
MICHEL	Frédéric	fmichel@notos.fr	04 30 96 97 32	

NOM	DATE D'INSERTION	TAILLE
Lago NoToS Ombre 100.jpg	20/03/2015	8 Ko
Lago NoToS Ombre 150.png	20/03/2015	11 Ko
Lago NoToS Ombre 211.png	20/03/2015	16 Ko
Lago NoToS Ombre 211 - Copie.png	20/03/2015	16 Ko
PUBLICATION.pdf	19/03/2015	94 Ko
50095d0147d2.pdf	18/03/2015	350 Ko



Pourquoi se tourner vers des solutions PHP+RPG

- Evolution des équipes IBM i
 - L'apprentissage facile de PHP et adapté aux développeurs RPG
 - Un script PHP peut ne contenir qu'une seule ligne de code
 - Pas besoin de tout connaître ! L'apprentissage est évolutif
 - RPG Free est comparable à PHP
 - L'objet n'est pas obligatoire : approche procédurale puis objet
 - Cohésion entre les équipes de développements
 - Rapprochement et implication des équipes IBM i et Web
 - Ressources développeurs web disponibles
 - Jeunes diplômés et autodidactes

Pourquoi se tourner vers des solutions PHP+RPG

→ Le meilleur des deux mondes

- RPG pour le langage de gestion

- Retourner la quantité en stock ou le tarif d'un article en suivant des règles métiers parfois complexes
- Enregistrement d'une commande client

- PHP pour les traitements à valeur ajoutée

- Les interfaces graphiques
- Construction d'Excel, de PDF avec fond de page
- Envoi de mails complexes (format HTML, pièce attachée, ...)
- Communication avec l'extérieur (Web Service, FTP, bases de données distantes)



Appeler du PHP depuis RPG

Appeler un script PHP depuis RPG

- Utile pour déléguer des tâches à un script PHP comme :
 - Convertir un spool produit par le *PGM RPG en PDF avec utilisation d'un fond de page, de logos, de codes à barres ou QR Code etc...
 - Enregistrement sur un stockage externe
 - Indexation pour accessibilité au travers d'un moteur de recherche

```

Fichier spoule
Fichier : : : : : FACTURE
Contrôle : : : : :
Recherche : : : : :
*.....1.....2.....3.....4.....5.....6.....7.....+
BRUNO TAVERNIER API DAYS 10102009A 51721 2009
API DAYS API DAYS
25, rue des merles 25, rue des merles
ZI des Milans ZI des Milans
34830 JACOU SUR CEZE 34830 JACOU SUR CEZE
01 02 03 04 05 01 02 03 04 05
ABRICOT PAMPRYL 031.89 0012 0000 031.89 0375
ADELSCOTT 024.29 0006 0000 024.29 0145
AMANDE EFFILE LITRE 1331.09 0013 0000 1331.09 16958
AMERICANO LITRE 029.99 0003 0000 029.99 0089
AMSTEL 149.87 0001 0000 149.87 0149
ANANAS BIB LITRE 016.27 0003 0000 016.27 0048
ANANAS LITRE 032.39 0007 0000 032.39 0226
ANANAS PAMPRYL 041.99 0007 0000 041.99 0293
ANIS GRAS LITRE 323.99 0003 0000 323.99 0971
ARMAGNAC LERICHE 323.99 0001 0000 323.99 0323
A suivre...
F9=Exit F12=Annuler F19=Gauche F20=Droite F24=Autres touches
    
```





10102009

FACTURE

Commercial
BRUNO TAVERNIER

Code Client
API DAYS 10102009A

Facture Date
N° **51721** 2009-09-12

Adresse de Facturation
API DAYS
25, rue des merles
ZI des Milans
34830 JACOU SUR CEZE
01 02 03 04 05

Adresse de Livraison
API DAYS
25, rue des merles
ZI des Milans
34830 JACOU SUR CEZE
01 02 03 04 05

Produit	PU HT	Qté	%	P. Net	Total Ligne
ABRICOT PAMPRYL	031.89	0012	0000	031.89	0375.07
ADELSCOTT	024.29	0006	0000	024.29	0145.76
AMANDE EFFILE LITRE	1331.09	0013	0000	1331.09	6958.14
AMERICANO LITRE	029.99	0003	0000	029.99	0089.98
AMSTEL	149.87	0001	0000	149.87	0149.87
ANANAS BIB LITRE	016.27	0003	0000	016.27	0048.81
ANANAS LITRE	032.39	0007	0000	032.39	0226.75
ANANAS PAMPRYL	041.99	0007	0000	041.99	0293.05
ANIS GRAS LITRE	323.99	0003	0000	323.99	0971.00
ARMAGNAC LERICHE	323.99	0001	0000	323.99	0323.00

Utilisation de QP2SHELL : côté RPG

- Côté RPG : appeler un script PHP avec passage de paramètres
- Appel du script sendmail.php avec passage d'une adresse mail en paramètre
- On peut aussi utiliser la syntaxe traditionnelle à base de CALL

```
..... /..1...+...2...+...3...+...4...+...5...+...6...+...7...+...8...
000100      D QP2Shell          PR          EXTPGM('QP2SHELL2')
000101      D  cmd              1024      const options(*varsize)
000102      D  parm1            1024      const options(*varsize:*nopass)
000103      D  parm2            1024      const options(*varsize:*nopass)
000105
000106      D  NULL              C          x'00'
000107      /FREE
000108
000109          QP2Shell('/usr/local/zendsvr6/bin/php-cli' + NULL :
000110                  '/www/zendsvr6/htdocs/IBM/sendmail.php' + NULL :
000111                  'dgayte@notos.fr' + NULL);
000115      *INLR = *ON;
000116      /END-FREE
```

QP2SHELL2

- A utiliser plutôt que QP2SHELL qui s'exécute dans un nouveau groupe d'activation
- Un Null Terminator (hexa 00) est nécessaire pour terminer chaque paramètre !

```
D NULL C x'00'
```

- La déclaration de ce type de procédure peut être dans un *include et intégré avec un /COPY*

```
/COPY BIB/QRPGLESRC, INC_PRC
```

Réception de paramètres en PHP

- Passage de paramètres traditionnels :
 - Entre deux scripts PHP : `$_GET`, `$_POST`, `$_SESSION`
 - Comment recevoir des paramètres en PHP depuis un programme RPG ?
- ➔ `argc` et `argv` : il faut activer la directive !
- `$argv` est un tableau de paramètres
 - `$argv[0]` est toujours le nom du script appelé
 - `$argv[n]` sont les paramètres qui sont transmis
 - `$argc` (argument count) est le nombre de paramètres
 - Vérification de la bonne réception des paramètres

```
1 <?php
2 if($argc) {
3     $destMail = trim($argv[1]);
```

Réception du paramètre : côté PHP

```
<?php
if($argc){
$destMail = trim($argv[1]);

file_put_contents('/home/gdumas/demo.txt', $destMail);

$headers = 'From: notos@notos.fr' . "\r\n" .
'Reply-To: notos@notos.fr' . "\r\n" .
'X-Mailer: PHP/' . phpversion();

mail($destMail, 'Démo IBM i PHP RPG', 'Succès de la démonstration',$headers);
}else{
file_put_contents('/home/gdumas/demo.txt', 'sans parm');
mail('gdumas@notos.fr', 'Démo IBM i PHP RPG', 'pas de parm reçu');
}
```

Rappel : Intégrateur

- Lance une commande sur le poste de travail
- Longueur de la commande passée au paramètre PCCMD
 - IBM i Access for Windows et Personal Communications toujours de 123
 - IBM i Access Client Solutions : 1023
- Fortement dépendant de l'OS du poste de travail

```
PGM      parm(&id)

DCL VAR(&CMD) TYPE(*CHAR) LEN(200)
DCL VAR(&PATH) TYPE(*CHAR) LEN(123)
DCL VAR(&id)   TYPE(*CHAR) LEN(5)

STRPCO PCTA(*NO)
MONMSG IWS4010

CHGVAR      VAR(&PATH) VALUE('"iexplore.exe" +
                           http://192.168.1.3/lorena/index.php?ent=crmibm&ht=' +
                           *TCAT &ID *TCAT '&dest=port')
CHGVAR VAR(&CMD) +
VALUE('rundll32 shell32,ShellExec_RunDLL ' *BCAT &PATH)

STRPCCMD PCCMD(&CMD) PAUSE(*NO)
ENDPGM
```



Appeler un programme RPG depuis PHP

Appeler un programme RPG sans paramètre

- Si le programme RPG n'attend pas de paramètre, nous pouvons l'appeler par une requête SQL en PHP par un CALL BIB.PGM

```
$cnx = db2_connect(HOST, USR, PWD);  
  
$requete = "CALL IBM2016.RENVOIPS2";  
  
$result = db2_exec($cnx, $requete);
```

- Permet de lancer un traitement RPG
- Possibilité de récupérer un ou plusieurs jeux de résultats (ResultSet)

Echanger des jeux de résultats

- Pour les anciennes versions de l'OS/400, la solution la plus utilisée pour échanger des jeux de résultats est l'écriture des enregistrements dans une table DB2

Exemple : retourner la liste des factures d'un client

1. Appel depuis PHP du programme RPG faisant la liste des factures du client en passant le numéro client en paramètre et un identifiant de la demande
 2. Le programme RPG écrit les résultats de la demande dans DB2
 3. Récupération des résultats en PHP par lecture de DB2
- ➔ Fonctionne techniquement mais nécessite le plus souvent de retoucher un peu les programmes RPG existants

Renvoyer un Result Set en RPG (SQL)

- Un curseur peut être renvoyé sous forme de Result Set
- Utiliser l'ordre SQL SET RESULT SETS Cursor C1

```
/FREE
EXEC SQL
    Declare C1 cursor      FOR
        SELECT CLIENT, NUMCDE, TOTAL
        FROM DGAYTE/ENTETE
        FOR FETCH ONLY    ;

EXEC SQL  Open C1;

EXEC SQL  SET RESULT SETS      Cursor C1;

*inlr = *on;
/END-FREE
```

RenvoiPS2

Renvoyer un Result Set en RPG (Tableau ou DS)

- Le Result Set renvoyé peut provenir d'un tableau ou de DS à occurrence multiple

```
d Ret          DS          DIM(3) QUALIFIED
d z1          6
d z2          5  2
d
/FREE
RET(1).Z1 = 'AAAAAA';
RET(2).Z1 = 'BBBBBB';
RET(3).Z1 = 'CCCCCC';

RET(1).Z2 = 123,45 ;
RET(2).Z2 = 234,56 ;
RET(3).Z2 = 345,67 ;

EXEC SQL
  SET RESULT SETS FOR RETURN TO CALLER
  array :Ret FOR 3 ROWS;
  *inlr = *on;
/END-FREE
```

RenvoiPST

Récupérer et traiter les jeux de résultats en PHP

- A exécuter comme une requête SQL DB2 classique

```
$cnx = db2_connect(HOST, USR, PWD);  
  
$requete = "CALL IBM2016.RENVOIPS3";  
  
$stmt = db2_exec($cnx, $requete);  
if(!$stmt) echo db2_stmt_errormsg();  
else while($row = db2_fetch_assoc($stmt))  
{  
    echo "  
        <tr>  
            <td>".$row['CLIENT']. "</td>  
            <td>".$row['NUMCDE']. "</td>  
            <td>".$row['TOTAL']. "</td>  
        </tr>  
        ";  
}
```

1. Connexion DB2

2. Appel du programme RPG

3. Parcours du premier jeu de résultats

Renvoyer plusieurs Result Set en RPG

- Pour renvoyer en une seule opération plusieurs listes de données

```
/FREE
EXEC SQL
  Declare C1 cursor      FOR
    SELECT CLIENT, NUMCDE, TOTAL
    FROM DGAYTE/ENTETE
    FOR FETCH ONLY  ;

EXEC SQL  Open C1;

EXEC SQL
  Declare C2 cursor      FOR
    SELECT  NUMCDE, count(*) AS NBCDE
    FROM DGAYTE/DETAIL
    GROUP BY NUMCDE
    FOR FETCH ONLY  ;

EXEC SQL  Open C2;

EXEC SQL  SET RESULT SETS  Cursor C1, CURSOR C2  ;

*inlr = *on;
/END-FREE
```

RenvoiPS3



Récupérer et traiter les jeux de résultats en PHP

```
$cnx = db2_connect(HOST, USR, PWD);  
  
$requete = "CALL IBM2016.RENVOIPS3";  
  
$stmt = db2_exec($cnx, $requete);  
if(!$stmt) echo db2_stmt_errormsg();  
else while($row = db2_fetch_assoc($stmt))  
{  
    echo "  
        <tr>  
            <td>".$row['CLIENT']. "</td>  
            <td>".$row['NUMCDE']. "</td>  
            <td>".$row['TOTAL']. "</td>  
        </tr>  
    ";  
}
```

1. Connexion DB2

2. Appel du programme RPG

3. Parcours du premier jeu de résultats

```
$stmt2 = db2_next_result($stmt);  
if(!$stmt2) echo db2_stmt_errormsg();  
else while($row2 = db2_fetch_assoc($stmt2))  
{  
    echo "  
        <tr>  
            <td>".$row2['NUMCDE']. "</td>  
            <td>".$row2['NBCDE']. "</td>  
        </tr>  
    ";  
}
```

4. Passage au prochain jeu de résultats

5. Parcours du second jeu de résultats

Appeler un programme RPG avec paramètres

- Plusieurs solutions :
 - Par procédures stockées externes
 - Par le Toolkit IBM i PHP
- Les exemples qui suivent se basent sur l'appel du programme RPG IBM2016/STOCK retournant la quantité en stock d'un article.

```
**
FSTOCKL      UF A E          K DISK      EXTFILE ('IBM2016/STOCKL')
DPDNBR              S              4
DPDQTT              S             10i 0
*
*'INITIALISATION
*' '
C
C      *ENTRY              PLIST
C              PARM
C              PARM              PDNBR
C              PDQTT
*
C              EVAL        PROD_NUM = PDNBR
C      PDNBR              CHAIN    INVENTORY
C              EVAL        PDQTT = QTY_IN_STK
C              SETON
```

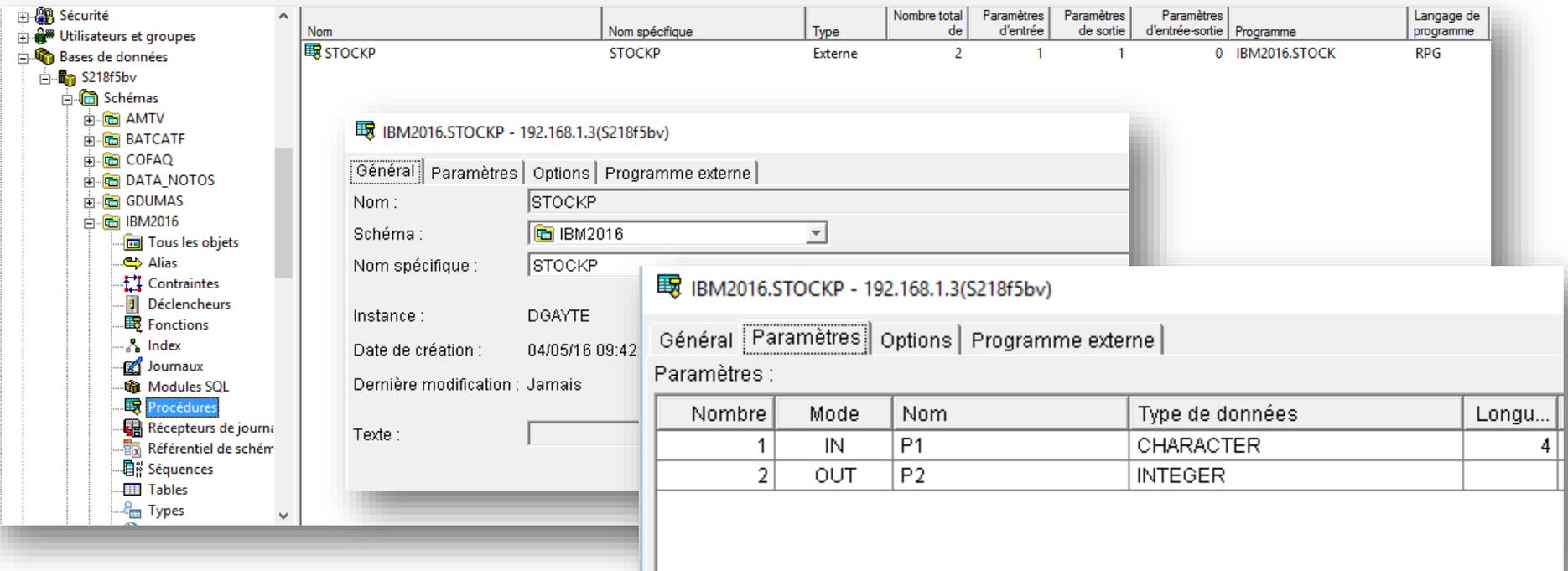
LR



Première solution : à l'aide des procédures stockées externes

Procédures stockées externes

- Permet d'appeler en SQL DB2 nos programmes RPG avec paramètres
- Création des procédures stockées externes par l'assistant de System i Navigator ou en SQL



The screenshot displays the IBM System i Navigator interface. On the left is a tree view of the database structure, including 'S218f5bv' and 'IBM2016'. The main window shows the configuration for the external stored procedure 'STOCKP'.

Nom	Nom spécifique	Type	Nombre total de	Paramètres d'entrée	Paramètres de sortie	Paramètres d'entrée-sortie	Programme	Langage de programme
STOCKP	STOCKP	Externe	2	1	1	0	IBM2016.STOCK	RPG

The configuration window for 'STOCKP' shows the following details:

- Nom :** STOCKP
- Schéma :** IBM2016
- Nom spécifique :** STOCKP
- Instance :** DGAYTE
- Date de création :** 04/05/16 09:42
- Dernière modification :** Jamais

The 'Paramètres' tab is active, showing the following table:

Nombre	Mode	Nom	Type de données	Longu...
1	IN	P1	CHARACTER	4
2	OUT	P2	INTEGER	

Procédures stockées externes

- La procédure stockée externe en SQL

```
CREATE PROCEDURE IBM2016.STOCKP (  
    IN P1 CHAR(4) CCSID 297 ,  
    OUT P2 INTEGER )  
LANGUAGE RPG  
SPECIFIC IBM2016.STOCKP  
NOT DETERMINISTIC  
MODIFIES SQL DATA  
CALLED ON NULL INPUT  
EXTERNAL NAME 'IBM2016/STOCK'  
PARAMETER STYLE GENERAL ;
```



Appeler une procédure stockée en PHP

```
$cnx = db2_connect(HOST, USR, PWD);  
  
//Initialisation des paramètres  
$PRODUCTNBR= '1002';  
$QTTE = 0;  
  
$cpro = db2_prepare($cnx,"CALL IBM2016.STOCKP( ? , ? ) ");  
  
db2_bind_param($cpro, 1, 'PRODUCTNBR', DB2_PARAM_IN);  
db2_bind_param($cpro, 2, 'QTTE', DB2_PARAM_OUT);  
  
if (db2_execute($cpro)) {  
    echo "<h3>Quantité en stock pour le  
        produit n° ".$PRODUCTNBR." : ".$QTTE."</h3>";  
}else {  
    echo db2_stmt_errormsg();  
}
```

1. Connexion DB2

2. Initialisation des paramètres

3. Préparation de l'appel de la PS

4. Binding des paramètres

5. Exécution et récupération du résultat



Deuxième solution : à l'aide du Toolkit PHP

Qu'est ce que le Toolkit ?

- Ensemble de classes PHP pour appeler les objets natifs IBM i
- Les programmes
 - RPG, COBOL, CL ...
- Autres :
 - Commandes IBM i
 - Spools, Job logs, Active jobs, Valeurs système
 - Data Area, User Spaces
- Accès naturel depuis le code PHP
- S'appuie sur la bibliothèque XMLSERVICE
- Plus d'informations sur :
<http://yips.idevcloud.com/wiki/index.php/XMLService/XMLSERVICE>

A l'aide du Toolkit PHP

- Pas de procédures stockées à maintenir
- Le développeur PHP peut se concentrer sur le code PHP

```
include_once 'ToolkitService.php';

$PRODUCTNBR= '1002';
$extension = 'ibm_db2';
try {
    //Connexion
    $ToolkitServiceObj = ToolkitService::getInstance(HOST,USR,PWD,$extension);

    $param[] = $ToolkitServiceObj->AddParameterChar('in', 4, 'PRODUCTNBR', 'PRODUCTNBR', $PRODUCTNBR);
    $param[] = $ToolkitServiceObj->AddParameterInt32('out', 'QTTE', 'QTTE', 0 );

    $result = $ToolkitServiceObj->PgmCall("STOCK","IBM2016", $param, null, null);

    if ($result){
        echo "<h3>Quantité en stock pour le produit n° ".$PRODUCTNBR." : ".$result['io_param']['QTTE']."</h3>";
    }
}

catch (Exception $e){
    echo $e->getMessage(), "\n";
    exit();
}
```



Exposer une logique RPG par Web Service

Les web services en PHP

- PHP est un langage prévu pour les web services
- Facile et rapide à déployer
- Normes SOAP ou REST

- Permet de partager des logiques métiers (*PGM) avec :
 - Des applications hors IBM i (réseau local ou internet)
 - Un site internet (gestion de stocks, de données clients)
 - D'autres interlocuteurs : clients, fournisseurs, partenaires...

- En conservant la sécurité :
 - Mise à disposition seulement des programmes choisis
 - Ouverture d'un port web et pas DB2

Les web services en PHP

- Avec les différentes techniques vues dans cette session, nous pouvons appeler un programme RPG en PHP sans difficulté

- Il ne reste plus qu'à encapsuler dans un web service ces appels :
 - 1. Ecrire la fonction PHP
 - 2. Déclarer le tableau retour du web service (équivalent d'une data structure)
 - 3. Déclarer la fonction PHP comme une méthode web service

- Pour l'exemple suivant, la bibliothèque PHP nuSoap est utilisée



1. Ecrire la fonction PHP

■ Fonction CallRenvoips2

```
function CallRenvoips2()
{
    include 'config.inc.php';

    $stab_result = array();

    $cnx = db2_connect(HOST, USR, PWD);

    $requete = "CALL IBM2016.RENVOIPS2";

    $result = db2_exec($cnx, $requete);
    if(!$result) echo db2_stmt_errormsg();
    else while($row = db2_fetch_assoc($result)){
        $stab_result[] = array(
            'CLIENT' => $row['CLIENT'],
            'NUMCDE' => $row['NUMCDE'],
            'TOTAL' => $row['TOTAL']
        );
    }

    return $stab_result;
}
```

1. Déclaration de la fonction

2. Appel du programme RPG

3. Retour des résultats

2. Déclaration du tableau retour

```
//Tableau retour pour renvoips2
$serv->wsdl->addComplexType(
    'Retour_Renvoips2',
    'complexType',
    'struct',
    'all', '',
    array(
        'CLIENT' => array('name' => 'CLIENT', 'type' => 'xsd:string'),
        'NUMCDE' => array('name' => 'NUMCDE', 'type' => 'xsd:string'),
        'TOTAL' => array('name' => 'TOTAL', 'type' => 'xsd:string'),
    )
);

$serv->wsdl->addComplexType(
    'Retour_Renvoips2Array',
    'complexType',
    'array',
    '',
    'SOAP-ENC:Array',
    array(),
    array(
        array('ref'=>'SOAP-ENC:arrayType', 'wsdl:arrayType'=>'tns:Retour_Renvoips2[]')
    ),
    'tns:Retour_Renvoips2'
);
```

Tableau à 3
colonnes :
CLIENT
NUMCDE
TOTAL

3. Déclaration de la méthode web service

```
$serv->register("CallRenvoips2",  
    array(),  
    array("Retour_Renvoips2"=>"tns:Retour_Renvoips2Array"),  
    'urn:CallRenvoips22',  
    'urn:CallRenvoips22/CallRenvoips2',  
    '',  
    '',  
    'Retourne un result set retourné par un programme RPG');
```



Exemple complet (1)

```
<?php
include 'nusoap/lib/nusoap.php';
$serv = new nusoap_server();
$serv->configureWSDL("SCORPION", "urn:SCORPION");
$serv->wsdl->schemaTargetNamespace="http://192.168.1.3/";

//Tableau retour pour renvoips2
$serv->wsdl->addComplexType(
    'Retour_Renvoips2',
    'complexType',
    'struct',
    'all','',
    array(
        'CLIENT' => array('name' => 'CLIENT', 'type' => 'xsd:string'),
        'NUMCDE' => array('name' => 'NUMCDE', 'type' => 'xsd:string'),
        'TOTAL' => array('name' => 'TOTAL', 'type' => 'xsd:string'),
    )
);

$serv->wsdl->addComplexType(
    'Retour_Renvoips2Array',
    'complexType',
    'array',
    '',
    'SOAP-ENC:Array',
    array(),
    array(
        array('ref'=>'SOAP-ENC:arrayType','wsdl:arrayType'=>'tns:Retour_Renvoips2[]')
    ),
    'tns:Retour_Renvoips2'
);
```



Exemple complet (2)

```
$serv->register("CallRenvoips2",
    array(),
    array("Retour_Renvoips2"=>"tns:Retour_Renvoips2Array"),
    'urn:CallRenvoips22',
    'urn:CallRenvoips22/CallRenvoips2',
    '',
    '',
    'Retourne un result set retourné par un programme RPG');

$serv->service($HTTP_RAW_POST_DATA);

function CallRenvoips2()
{
    include 'config.inc.php';

    $stab_result = array();

    $cnx = db2_connect(HOST, USR, PWD);

    $requete = "CALL IBM2016.RENVOIPS2";

    $result = db2_exec($cnx, $requete);
    if(!$result) echo db2_stmt_errormsg();
    else while($row = db2_fetch_assoc($result)){
        $stab_result[] = array(
            'CLIENT' => $row['CLIENT'],
            'NUMCDE' => $row['NUMCDE'],
            'TOTAL' => $row['TOTAL']
        );
    }

    return $stab_result;
}
```

Accessible depuis un navigateur

- Description accessible depuis un navigateur



SCORPION

View the [WSDL](#) for the service. Click on an details.

[CallRenvoips2](#)

[Close](#)

Name: CallRenvoips2
Binding: SCORPIONBinding
Endpoint: http://192.168.1.3/IBM/modernisation2016/rpg_ws.php
SoapAction: um:CallRenvoips22/CallRenvoips2
Style: rpc

Input:
use: encoded
namespace: um:CallRenvoips22
encodingStyle: <http://schemas.xmlsoap.org/soap/encoding/>
message: CallRenvoips2Request
parts:

Output:
use: encoded
namespace: um:CallRenvoips22
encodingStyle: <http://schemas.xmlsoap.org/soap/encoding/>
message: CallRenvoips2Response
parts:
Retour_Renvoips2: tns:Retour_Renvoips2Array

Namespace: um:CallRenvoips22
Transport: <http://schemas.xmlsoap.org/soap/http>
Documentation: Retourne un result set retourné par un programme RPG

Merci pour votre écoute !

Des questions ?

S45 - Les différentes techniques d'intégration PHP + RPG

Gautier DUMAS - gdumas@notos.fr
Dominique GAYTE – dgayte@notos.fr
04 30 96 97 31
www.notos.fr

