



USING A WEB SERVICE FOR INTERCHANGEABLE DATA ACCESS

Presented by Jay A. Jolicoeur
Principal Software Architect
Switchback, Inc.

THE PRESENTATION

The presentation is intended to show how a web service can be used to access data on a server regardless of the database used or storage location.

The server side is written in PHP, and for the purposes of this presentation the client side is also written in PHP, although the client side is not limited to a specific language.

Complete server side code is included with examples for accessing MySQL and MS SQL data.

The web server on the service side is Apache 2.4.18 with PHP 5.6.21, cURL 7.43.0, MySQL 5.0.11. mssql with FreeTDS 1.00

Checkout the web service source from <http://www.switchback.com/code/SwDataWebService.zip>.

Sample client source code can be found at <http://www.switchback.com/code/WebServiceClientSample.zip>.



THE CLIENT ACTIONS

There are only 2 client actions:

- retrieve – retrieve data from the web service
- post – posting data to the web service for insert, update or delete

All these actions are performed using the `WebservClient` class as found in `class_webserv_client.php` and is included as part of this presentation.

See the sample client code.



RETRIEVING DATA

The client retrieves data from the web service using one of 3 methods

- Retrieve by keys (BY_KEY)
- Retrieve by resource id
- Retrieve by query (BY_QUERY)

All retrieval methods, except BY_QUERY, use a set of key identifiers to indicate the column names and values to be used for the retrieval.

Key Identifiers include

p_table	The name of the table to retrieve data from
k_key_id_x	The column used for the retrieval, replace the 'x' with the key number in sequence (1-n)
k_key_oper_x	The operation (=, >, <, <>,) for the key, if not used '=' is assumed.
k_key_val_x	The key value
k_rtn_cols	The columns to return, if not used, all columns from the table will be returned
k_order	The columns used to sort the query, if not used, the key columns will be used in the sort. Can include 'DESC', 'ASC', 'LIMIT X'. For MS SQL queries, LIMIT is changed to TOP on the server side.



RETRIEVING DATA BY KEYS

Retrieving by keys (BY_KEY) is used to retrieve data from a single table.

Example:

```
$params = [  
  'p_table'      => 'table1',  
  'k_key_id_1'   => 't1_coll',  
  'k_key_oper_1' => '=',  
  'k_key_val_1'  => $coll_value,  
  'k_rtn_cols'   => 't1_coll,t1_col2,t1_col3',  
  'k_order'      => 't1_col2 DESC'];  
$tm = new \WebsrvClient;  
$tm->retrieve(BY_KEY, SW_WSGET, $params);
```



RETRIEVING DATA BY KEYS

Retrieving by keys (BY_KEY) is used to retrieve data from a single table.

Alternate method:

```
$tm = new \WebsrvClient;  
$tm->setTable('table1');  
$tm->setKeyld(1, 't1_col1');  
$tm->setKeyOper(1, '=');  
$tm->setKeyVal(1, $coll_value);  
$tm->setRtnCols('t1_col1, t1_col2, t1_col3');  
$tm->setOrder('t1_col2 DESC');  
$tm->retrieve(BY_KEY, SW_WSGET);
```



RETRIEVING DATA BY RESOURCE ID

Retrieving by Resource ID is used when the data to be retrieved is from multiple tables using a pre-defined query that resides on the webservice server.

Example client code:

```
$tm = new \WebsrvClient;  
$tm->setKeyOper('=');  
$tm->setKeyVal($coll_value);  
$tm->retrieve('resource_one', SW_WSGET);
```

Predefined resource 'resource_one' on the server:

```
SELECT t1_coll, t1_col2, t2_col2  
FROM table1 WITH (NOLOCK)  
INNER JOIN table2 WITH (NOLOCK)  
ON t2_coll = t1_coll  
WHERE t1_coll {_key_oper_1} {_key_val_1}
```



RETRIEVING DATA BY QUERY

To retrieve by query, simply include the complete query.

```
$query = "SELECT t1_col1, t1_col2, t2_col2  
FROM table1 WITH (NOLOCK)  
INNER JOIN table2 WITH (NOLOCK)  
ON t2_col1 = t1_col1  
WHERE t1_col1 = '$col1_value'";
```

Example client code:

```
$tm = new \WebsrvClient;  
$tm->retrieve(BY_QUERY, SW_WSGET, $query);
```



IDENTIFYING A TABLE

The web service

```
public static $valid_tables = [  
  'dd_apps'      => ['HOST' => 'S', 'DB' => 'structure', 'SCHEMA' => 'swconfig.', 'CONNECT' => 'userid:pwd'],  
  'dd_programs' => ['HOST' => 'S', 'DB' => 'structure', 'SCHEMA' => 'swconfig.', 'CONNECT' => 'userid:pwd'],  
  'dd_tables'   => ['HOST' => 'S', 'DB' => 'structure', 'SCHEMA' => 'swconfig.', 'CONNECT' => 'userid:pwd'],  
  'dd_fields'   => ['HOST' => 'S', 'DB' => 'structure', 'SCHEMA' => 'swconfig.', 'CONNECT' => 'userid:pwd'],  
  'dd_listmaster' => ['HOST' => 'S', 'DB' => 'structure', 'SCHEMA' => 'swconfig.', 'CONNECT' => 'userid:pwd'],  
  'table_dd_def' => ['HOST' => 'S', 'DB' => 'structure', 'SCHEMA' => '', 'CONNECT' => 'userid:pwd'],  
  'cas_requests' => ['HOST' => 'S', 'DB' => 'cas_admin', 'SCHEMA' => '', 'CONNECT' => 'userid:pwd'],  
  'cas_requests_detail' => ['HOST' => 'S', 'DB' => 'cas_admin', 'SCHEMA' => '', 'CONNECT' => 'userid:pwd'],  
  '/* S */'      => ['HOST' => 'S', 'DB' => 'structure', 'SCHEMA' => '', 'CONNECT' => 'userid:pwd'],  
  '/* M */'      => ['HOST' => 'M', 'DB' => 'castime', 'SCHEMA' => ''],  
  'rtu_srvr_emptime' => ['HOST' => 'S', 'DB' => 'cas_rtu', 'SCHEMA' => ''],  
  'rtu_devices'  => ['HOST' => 'S', 'DB' => 'cas_rtu', 'SCHEMA' => '']  
];
```



ON THE WEB SERVICE SIDE

The web service uses database classes created by Switchback, Inc., that simplify the access of any database, without having to know the specifics of PDO, mysqli, or mssql. Depending on the environment or version of PHP, either of the device dependent drivers can be used but the data access class objects remain the same.



CONTINUATION ...

To be continued

