

## **SOAP Web Services**

### **What is a SOAP web service**

Used primarily as a means for businesses to communicate with each other and with clients. Web services allow organizations to **exchange** data without intimate knowledge of **each other's** IT systems behind the firewall.

A technical description from Wikipedia:

*"A 'Web service' (also Web Service) is defined by the W3C as "a software system designed to support interoperable machine-to-machine interaction over a network".[1] Web services are frequently just Web APIs that can be accessed over a network, such as the Internet, and executed on a remote system hosting the requested services.*

*The W3C Web service definition encompasses many different systems, but in common usage the term refers to clients and servers that communicate over the HTTP protocol used on the Web...*

In general, a SOAP message could be sent to a web service enabled web site, for example, a house price database, with the parameters needed for a search. The site will return the resulting data (prices, location, features, etc) as an XML-formatted document. The SOAP client parses the XML so the data is formatted in a standardized machine-parseable format so it may be integrated directly into your site.

To put it in Filemaker development terms, it's like calling a plug-in function, but a plug-in that is called over the internet and that can perform an action on the remote server (like a financial transaction) and/or return useful data to you (like the distance example given).

### **Why would someone use a web service**

Setup a WSDL server to provide data for your clients without having to give them direct access to your database or you may need to create a client application to access data provided by a partner's web service. The example given here could become a store locator using zip codes.

### **What does this one do?**

This example calculates the distance between zip codes. The zip codes are used to perform a lookup on a web services (wsdl) server and uses the longitude and latitude of each zip code to determine the distance between locations and the city, and state of each zip code.

## Sample zip code output

### Distance between 92028 and 92589

Zip code 92028 is 12.01 miles away from 92589

#### Zip Code Details for 92028

Latitude: 33.3687  
Longitude: -117.219  
City: Fallbrook  
County: San Diego  
State Prefix: CA  
State Name: California  
Area Code: 760  
Time Zone: Pacific

#### Zip Code Details for 92589

Latitude: 33.517  
Longitude: -117.11  
City: Temecula  
County: Riverside  
State Prefix: CA  
State Name: California  
Area Code: 909  
Time Zone: Pacific

[try another](#)

## Development process.

The development process started with simplifying the problem. This involved developing the PHP/MySQL script traditionally as a single script. This was done by coding the forms, validation, functionality. Once this was working, the 'Lookup Code' was moved into a WSDL server script and original script became the client by creating an instantiate of the NuSOAP (a PHP library that performs the SOAP action) object and calling the server function from the client.

The main issue, aside from the coding, was to determine the location of the WSDL server.

The beauty of any Web Service is that there can be many clients installed on many computers talking to only one WSDL server. The server is where the main functionality lives and having it in only one location makes fixing the code and adding functionality much easier.

## Real world use for this kind of service.

This web service could be used to lookup any relevant zip code data. Some uses include: finding the nearest company store, to help calculate shipping costs, as input for google maps.

## Sample code:

### Client code snippet

```
require_once 'lib/nusoap.php';

$current_page = $_SERVER['HTTP_HOST'] . dirname($_SERVER['PHP_SELF']) ;

$url = 'http://<url to SOAP server/distance_server.php';

// Create the client instance
$client = new nusoap_client($url);
// Check for an error
$error = $client->getError();
if ($error) {
    // Display the error
    echo '<p><b>Constructor error: ' . $error . '</b></p>';
    // At this point, you know the call that follows will fail
}
// Call the SOAP method
list($distance,$zip1_summary,$zip2_summary) = $client->call('distance', array('zip1' => $zip1,'zip2' => $zip2));

// Display the request and response
/*echo '<h2>Request</h2>';
echo '<pre>' . htmlspecialchars($client->request, ENT_QUOTES) . '</pre>';
echo '<h2>Response</h2>';
echo '<pre>' . htmlspecialchars($client->response, ENT_QUOTES) . '</pre>';

// Display the debug messages
echo '<h2>Debug</h2>';
echo '<pre>' . htmlspecialchars($client->debug_str, ENT_QUOTES) . '</pre>';
```

### Server code snippet

```
<?php
// Pull in the NuSOAP code
require_once '../lib/nusoap.php';
require_once '../lib/zipcode.class.php';
// Enable debugging *before* creating server instance
$debug = 1;
// Define the method as a PHP function
function distance ($zip1,$zip2) {
    //connect to db

    mysql_connect('localhost',<username>,'<password>') or die(mysql_error());
    mysql_select_db(<dbname>) or die(mysql_error());

    $z = new zipcode_class;
    $miles = $z->get_distance($zip1,$zip2);

    if ($miles === false) {
        $distance = 'Error: '.$z->last_error;
    } else {
```

*Server code snippet (cont.)*

```
        $distance = "<span style='display: block; text-align: center; margin-top: 30px; padding: 0;'>Zip code <b>$zip1</b> is
        <b>$miles</b> miles away from <b>$zip2</b><span>r";
    }
}
```

```

$zip1_details = $z->get_zip_details($zip1);

if ($zip1_details === false) {
    $zip1_summary = ""; //Error: '$z->last_error';
} else {
    $zip1_summary .= "<p style='float: left; margin-left:145px; margin-top: 35px; padding: 0;'><strong>Zip Code Details for
$zip1</strong><br /><br />";
    foreach ($zip1_details as $key => $value) {
        $key = str_replace('_', '$key');
        $key = ucwords($key);
        $zip1_summary .= "$key:&nbsp;$value<br />";
    }
    $zip1_summary .= "</span>";
}

$zip1_details = $z->get_zip_details($zip2);

if ($zip1_details === false) {
    $zip2_summary = ""; //Error: '$z->last_error';
} else {
    $zip2_summary .= "<p style='float: center; margin-left:375px; margin-top: 35px; padding: 0;'><strong>Zip Code Details
for $zip2</strong><br /><br />";
    foreach ($zip1_details as $key => $value) {
        $key = str_replace('_', '$key');
        $key = ucwords($key);
        $zip2_summary .= "$key:&nbsp;$value<br />";
    }
    $zip2_summary .= "</span>";
}

return array($distance,$zip1_summary,$zip2_summary);
}
// Create the server instance
$server = new nusoap_server;
// Register the method to expose
$server->register('distance');

// Use the request to (try to) invoke the service
$HTTP_RAW_POST_DATA = isset($HTTP_RAW_POST_DATA) ? $HTTP_RAW_POST_DATA : "";
$server->service($HTTP_RAW_POST_DATA);
?>

```

## Conclusion.

SOAP web services using PHP and the nuSOAP library is a great way for a vendor to provide their customers a way to retrieve specific data such as order information, store locations or anything that is stored in the vendors database. Each customer can develop their own WSDL client to retrieve just the information needed to keep their office database up-to-date. No more sending spreadsheets and manual data entry needed. It's as if your vendor's database in another machine on your network.