

Example 1 – PHP Web Service:

- Make a virtual host
- e.g. `http://api.local`
- Don't forget to restart apache
- Add an entry to your hosts file

```
<VirtualHost *:80>
    ServerName api.local
    ServerAdmin admin@localhost
    DocumentRoot /var/www/myapi/public
    <Directory /var/www/myapi/public>
        AllowOverride All
        Order deny,allow
        Allow from All
    </Directory>
</VirtualHost>
```

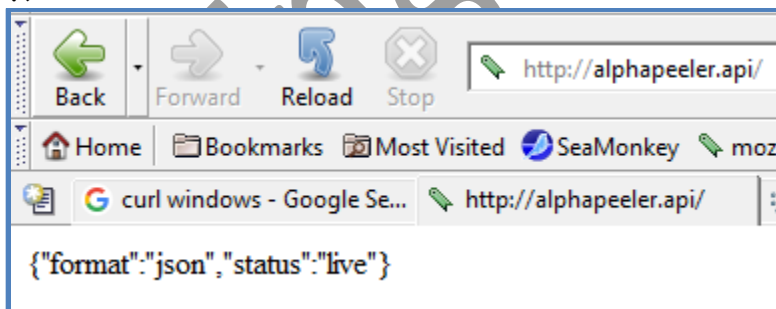
```
<VirtualHost *:80>
    ServerName alphapeeler.api
    ServerAdmin admin@localhost
    DocumentRoot "D:/xampp/htdocs/alphapeeler"
    <Directory "D:/xampp/htdocs/alphapeeler">
        AllowOverride All
        Order deny,allow
        Allow from All
    </Directory>
</VirtualHost>
```

Create the index.php file

- e.g. `D:/xampp/htdocs/alphapeeler/index.php`

Example 1: index.php

```
<?php
$data = array(
    'format' => 'json',
    'status' => 'live'
);
echo json_encode($data);
?>
```

**Use cURL from the command line**

- `curl http://alphapeeler.api`
- For more information about curl:

<https://curl.haxx.se/download.html>

https://winampplugins.co.uk/download.php?file=curl/curl_7_53_1_openssl_nghttp2_x64.7z

https://winampplugins.co.uk/download.php?file=curl/curl_7_53_1_openssl_nghttp2_x86.7z

JSON JavaScript Object Notation

- Originally for JavaScript
- Native read/write in most languages
- Simple, lightweight format - useful for mobile
- In PHP we have `json_encode()` and `json_decode()`
- These work with arrays and objects

Our service returns:

```
{'format':'json','status':'live'}
```

Heartbeat Method

- A method which does nothing
- No authentication
- Requires correct request format
- Gives basic feedback
- Shows that service is alive

Delivering A Web Service

- Service
- Documentation
- Examples
- A help point

HTTP and Data Formats

HTTP

HTTP is Hypertext Transfer Protocol. It is designed to exchange information about a request/response.

- Status Codes (e.g. 200, 404)
- Headers (e.g. Content-Type, Authorization)
- Verbs (e.g. GET, POST)

Status Codes

Win or fail? Some common codes:

200 OK

204 No Content

302 Found

301 Moved Permanently

302 Found

400 Bad Request

401 Not Authorised

403 Forbidden

404 Not Found

500 Internal Server Error

Working with Status Codes in PHP

We can observe status codes with curl, passing the `-I` switch

```
curl -I http://alphapeeler.api
```

Let's amend our web service, to return a 302 header

Example 2: index.php

```
header("302 Found", true, 302);
```

```
$data = array(  
'format' => 'json',  
'status' => 'live'  
);
```

```
echo json_encode($data);
```

HTTP Verbs

- More than GET and POST
- PUT and DELETE to update and delete in a RESTful service
- HEAD, OPTIONS and others also specified

In REST, we use:

GET Read

POST Create

PUT Update

DELETE Delete

HTTP Headers

Headers are the metadata about the content we send/receive

Useful headers:

- Accept and Content-Type: used for content format negotiation
- User-Agent: to identify what made the request
- Set-Cookie and Cookie: working with cookie data
- Authorization: controlling access

Accept Header

What type of content can the consumer understand?

- -v with curl to see request and response headers
- -H to add headers

```
curl -v -H "Accept: text/html" http://alphapeeler.api
```

Gives the output:

```
curl -v -H "Accept: text/html" http://alphapeeler.api
```

```
* Rebuilt URL to: http://alphapeeler.api/
```

```
* Trying 127.0.0.1...
```

```
* TCP_NODELAY set
```

```
* Connected to alphapeeler.api (127.0.0.1) port 80 (#0)
```

```
> GET / HTTP/1.1
```

```
> Host: alphapeeler.api
```

```
> User-Agent: curl/7.53.1
```

```
> Accept: text/html
```

```
>
```

```
< HTTP/1.1 302 Found
```

```
< Date: Mon, 17 Apr 2017 23:18:16 GMT
```

```
< Server: Apache/2.4.3 (Win32) OpenSSL/1.0.1c PHP/5.4.7
```

```
< X-Powered-By: PHP/5.4.7
```

```
< Content-Length: 33
```

```
< Content-Type: text/html
```

```
<
```

```
{"format":"json","status":"live"}* Connection #0 to host alphapeeler.api left intact
```

```
F:\FAST\webprog\myslides\week12\curl32>
```

```

C:\Windows\system32\cmd.exe
F:\FAST\webprog\myslides\week12\curl32>curl -v -H "Accept: text/html" http://alphapeeler.api
* Rebuilt URL to: http://alphapeeler.api/
* Trying 127.0.0.1...
* TCP_NODELAY set
* Connected to alphapeeler.api (127.0.0.1) port 80 (#0)
> GET / HTTP/1.1
> Host: alphapeeler.api
> User-Agent: curl/7.53.1
> Accept: text/html
< HTTP/1.1 302 Found
< Date: Mon, 17 Apr 2017 23:18:16 GMT
< Server: Apache/2.4.3 (Win32) OpenSSL/1.0.1c PHP/5.4.7
< X-Powered-By: PHP/5.4.7
< Content-Length: 33
< Content-Type: text/html
{"format":"json","status":"live"}* Connection #0 to host alphapeeler.api left intact
F:\FAST\webprog\myslides\week12\curl32>

```

Using the Accept Header

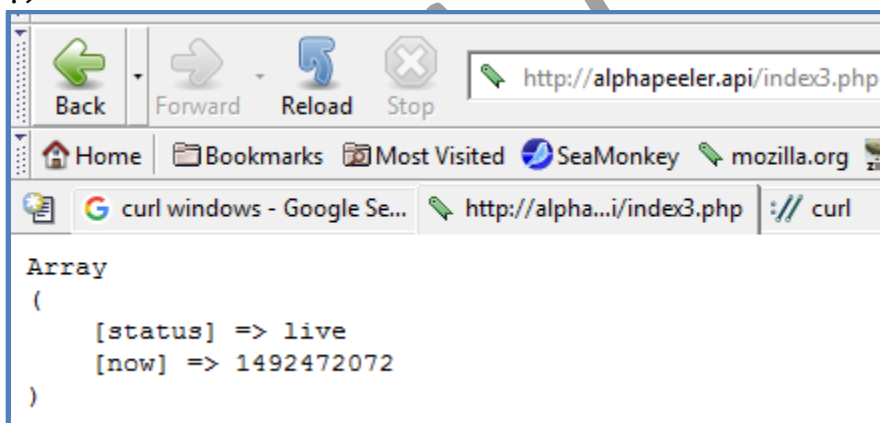
We can work out what format the user wanted to see from the Accept header.

Example 3: index3.php

```

<?php
$data = array(
    'status' => 'live',
    'now' => time()
);
if (false !== strpos($_SERVER['HTTP_ACCEPT'], 'text/html')) {
    echo "<pre>";
    print_r($data);
    echo "</pre>";
} else {
    // return json
    echo json_encode($data);
}
?>

```



```

curl -v -H "Accept: text/html" http://alphapeeler.api/index3.php
* Trying 127.0.0.1...
* TCP_NODELAY set
* Connected to alphapeeler.api (127.0.0.1) port 80 (#0)
> GET /index3.php HTTP/1.1
> Host: alphapeeler.api
> User-Agent: curl/7.53.1
> Accept: text/html

```

```
>
< HTTP/1.1 200 OK
< Date: Mon, 17 Apr 2017 23:35:30 GMT
< Server: Apache/2.4.3 (Win32) OpenSSL/1.0.1c PHP/5.4.7
< X-Powered-By: PHP/5.4.7
< Content-Length: 66
< Content-Type: text/html
<
<pre>Array
(
    [status] => live
    [now] => 1492472130
)
</pre>* Connection #0 to host alphapeeler.api left intact
F:\FAST\webprog\myslides\week12\curl32>
```

More examples - Handle Accept Headers

Example accept header (from my browser)

text/html, application/xml;q=0.9, application/xhtml+xml,
image/png, image/jpeg, image/gif, image/x-xbitmap, */*;q=0.1

Content-Type Header

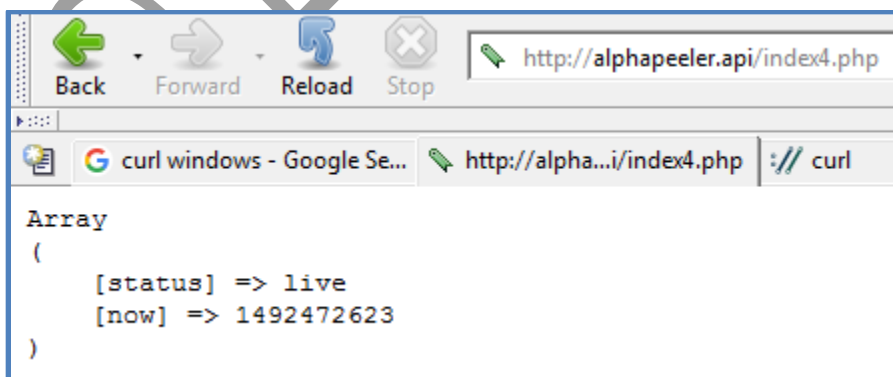
The Content-Type header: literally labels the contents of the response.

We can include these in our examples:

Example 4: index4.php

```
<?php
$data = array(
    'status' => 'live',
    'now' => time()
);

if (false !== strpos($_SERVER['HTTP_ACCEPT'], 'text/html')) {
    echo "<pre>";
    print_r($data);
    echo "</pre>";
} else {
    // return json
    header('Content-Type: application/json');
    echo json_encode($data);
}
?>
```



```
curl -v http://alphapeeler.api/index4.php
* Trying 127.0.0.1...
* TCP_NODELAY set
* Connected to alphapeeler.api (127.0.0.1) port 80 (#0)
> GET /index4.php HTTP/1.1
> Host: alphapeeler.api
> User-Agent: curl/7.53.1
> Accept: */*
>
< HTTP/1.1 200 OK
< Date: Mon, 17 Apr 2017 23:44:54 GMT
< Server: Apache/2.4.3 (Win32) OpenSSL/1.0.1c PHP/5.4.7
< X-Powered-By: PHP/5.4.7
< Content-Length: 34
< Content-Type: application/json
<
{"status":"live","now":1492472694}* Connection #0 to host alphapeeler.api left i
ntact
```

```
F:\FAST\webprog\myslides\week12\curl32>
```

Adding XML to Our Service

Example 5: index5.php

```
<?php
$data = array(
    'status' => 'live',
    'now' => time()
);

if (false !== strpos($_SERVER['HTTP_ACCEPT'], 'text/html')) {
    echo "<pre>";
    print_r($data);
    echo "</pre>";
} elseif (false !== strpos($_SERVER['HTTP_ACCEPT'], 'text/xml')) {
    $simplexml = simplexml_load_string('<?xml version="1.0" ?><data />');
    foreach($data as $key => $value) {
        $simplexml->addChild($key, $value);
    }
    header('Content-Type: text/xml');
    echo $simplexml->asXML();
} else {
    // return json
    header('Content-Type: application/json');
    echo json_encode($data);
}
?>
```

```
curl -v -H "Accept: text/xml" http://alphapeeler.api/index5.php
* Trying 127.0.0.1...
* TCP_NODELAY set
* Connected to alphapeeler.api (127.0.0.1) port 80 (#0)
```

```
> GET /index5.php HTTP/1.1
> Host: alphapeeler.api
> User-Agent: curl/7.53.1
> Accept: text/xml
>
< HTTP/1.1 200 OK
< Date: Tue, 18 Apr 2017 00:04:45 GMT
< Server: Apache/2.4.3 (Win32) OpenSSL/1.0.1c PHP/5.4.7
< X-Powered-By: PHP/5.4.7
< Content-Length: 78
< Content-Type: text/xml
<
<?xml version="1.0"?>
<data><status>live</status><now>1492473885</now></data>
* Connection #0 to host alphapeeler.api left intact
```

F:\FAST\webprog\myslides\week12\curl32>

Versioning

- Always include a version parameter or media type
- ```
http://example.com/api/v4/status
http://example.com/api/status
Content-Type: application/vnd.myapi.2+json
```

### Data Formats

- Handle multiple formats, by header or parameter
  - JSON
  - XML
  - ?
  - Common to detect header, allow parameter override
- ```
http://example.com/api/v4/status
Accept: application/json
http://example.com/api/v4/status?format=json
Accept: text/html
```

Statelessness

- Request alone contains all information needed
- No data persistence between requests
- Resource does not need to be in known state
- Same operation performs same outcome