



USING HAPI PLUGINS TO VERSION YOUR API

Dave Stevens - zappos.com

@davestevens84 | dstevens@zappos.com



WHO?

Because fame !== true

- Started using node in 2010
- First used hapi fall of 2013
- 6pm.com mobile web in production powered by hapi late 2013 (test)
- hapi 8 in prod today (internal analytics API)



PROJECT SETUP

Let's keep this brief

```
/
|_ index.js
|_ lib/
|   |_ index.js
|   |_ modules/
|       |_ info/
|           |_ index.js
|           |_ package.json
|           |_ version/
|               |_ index.js
|_ package.json
```

Full details of project organization:

<http://bit.ly/orghapi>



MAIN APPLICATION

```
1  var Glue = require('glue');
2
3  var manifest = {
4    connections: [
5      {
6        port: 9000
7      }
8    ],
9    plugins: {
10     './info': null
11   }
12 };
13
14 var options = {
15   relativeTo: process.cwd() + '/lib/modules'
16 };
17
18 Glue.compose(manifest, options, function (err, server) {
19   if (err) throw err;
20
21   server.start(function (err) {
22     console.log('Server running on port 9000');
23   });
24 }
25 );
```



INFO PLUGIN

```
1  exports.register = function (server, options, next) {  
2  
3      server.route({  
4          path: '/version',  
5          method: 'GET',  
6          handler: require('./version')  
7      });  
8  
9      next();  
10  
11  };  
12  
13  exports.register.attributes = {  
14      pkg: require('./package.json')  
15  };
```



VERSION HANDLER

```
1  module.exports = function (request, reply) {  
2  
3      reply({  
4          name: "exampleAPI",  
5          version: "1"  
6      });  
7  
8  };
```



TEST IT OUT

```
$ curl http://localhost:9000/version
```

```
{  
  "name": "exampleAPI",  
  "version": "1"  
}
```



HERE COMES VERSION 2

and it'll introduce some breaking changes



VERSION HANDLER

Before

```
|_ modules/  
  |_ info/  
    |_ index.js  
    |_ package.json  
    |_ version/  
      |_ index.js
```

After

```
|_ modules/  
  |_ info/  
    |_ index.js  
    |_ package.json  
    |_ version/  
      |_ index.js  
      |_ v1/  
        |_ index.js  
      |_ v2/  
        |_ index.js
```



VERSION HANDLER - V1

```
$ cd version  
$ mkdir v1  
$ mv index.js v1/index.js
```

```
1  module.exports = function (request, reply) {  
2  
3      reply({  
4          name: "exampleAPI",  
5          version: "1"  
6      });  
7  
8  };
```



VERSION HANDLER - V2

```
$ cd version  
$ mkdir v2
```

```
1  module.exports = function (request, reply) {  
2  
3      reply({  
4          statusCode: 200,  
5          applicationDetail: {  
6              name: "exampleAPI",  
7              version: "2"  
8          }  
9      });  
10  
11  };
```



VERSION HANDLER - INDEX

```
1  module.exports = function (request, reply) {  
2  
3      if (request.params.apiVersion && (/^v[0-9]+$/.test(request.params.apiVersion) ) {  
4          return require('./' + request.params.apiVersion)(request, reply);  
5      }  
6  
7      return require('./v1')(request, reply);  
8  
9  };
```



INFO PLUGIN - INDEX

```
1  exports.register = function (server, options, next) {  
2  
3      server.route({  
4          path: '/version',  
5          method: 'GET',  
6          handler: require('./version')  
7      });  
8  
9      server.route({  
10         path: '/{apiVersion}/version',  
11         method: 'GET',  
12         handler: require('./version')  
13     });  
14  
15     next();  
16  
17 };  
18  
19 exports.register.attributes = {  
20     pkg: require('./package.json')  
21 };
```



VERSION HANDLER - INDEX

```
1  module.exports = function (request, reply) {  
2  
3      if (request.params.apiVersion && (/^v[0-9]+$/.test(request.params.apiVersion) ) {  
4          return require('./' + request.params.apiVersion)(request, reply);  
5      }  
6  
7      return require('./v1')(request, reply);  
8  
9  };
```



VERSION HANDLER - INDEX

```
$ curl http://localhost:9000/version
```

```
{  
  "name": "exampleAPI",  
  "version": "1"  
}
```



VERSION HANDLER - INDEX

```
$ curl http://localhost:9000/v1/version
```

```
{  
  "name": "exampleAPI",  
  "version": "1"  
}
```




VERSION HANDLER - INDEX

```
$ curl http://localhost:9000/v2/version
```

```
{  
  "statusCode":200,  
  "applicationDetail":{  
    "name":"exampleAPI",  
    "version":"2"  
  }  
}
```



VERSIONING WITH CUSTOM HEADER

<http://bit.ly/troyhuntapi>

```
1  module.exports = function (request, reply) {
2
3      // Versioned by URI route
4      if (request.params.apiVersion && (/^v[0-9]+$/.test(request.params.apiVersion) ) {
5          return require('./' + request.params.apiVersion)(request, reply);
6      }
7
8      // Versioned by customer request header
9      if (request.headers['api-version'] && (/^[0-9]+$/.test(request.headers['api-version'])) ) {
10         return require('./v' + request.headers['api-version'])(request, reply);
11     }
12
13     // Default gets original API, to be backwards compatible
14     return require('./v1')(request, reply);
15
16 };
```



VERSIONING WITH CUSTOM HEADER

<http://bit.ly/troyhuntapi>

```
$ curl -H "api-version: 1"  
http://localhost:9000/version
```

```
{  
  "name": "exampleAPI",  
  "version": "1"  
}
```



VERSIONING WITH CUSTOM HEADER

<http://bit.ly/troyhuntapi>

```
$ curl -H "api-version: 2"  
http://localhost:9000/version
```

```
{  
  "statusCode":200,  
  "applicationDetail":{  
    "name":"exampleAPI",  
    "version":"2"  
  }  
}
```



VERSIONING WITH ACCEPT HEADER

<http://bit.ly/troyhuntapi>

```
1 module.exports = function (request, reply) {
2
3   // Versioned by URI route
4   if (request.params.apiVersion && (/^v[0-9]+$/.test(request.params.apiVersion) ) {
5     return require('./' + request.params.apiVersion)(request, reply);
6   }
7
8   // Versioned by customer request header
9   if (request.headers['api-version'] && (/^[0-9]+$/.test(request.headers['api-version'])) ) {
10    return require('./v' + request.headers['api-version'])(request, reply);
11  }
12
13  // Versioned by accept header
14  if (request.headers['accept'] && (/^application/vnd.exampleapi.v[0-9]+$/.test(request.headers['accept'])) ) {
15
16    var version = request.headers['accept'].replace('application/vnd.exampleapi.', '');
17
18    return require('./' + version)(request, reply);
19  }
20
21  // Default gets original API, to be backwards compatible
22  return require('./v1')(request, reply);
23
24  };
```



VERSIONING WITH ACCEPT HEADER

<http://bit.ly/troyhuntapi>

```
$ curl -H "accept: application/vnd.exampleapi.v1"  
      http://localhost:9000/version
```

```
{  
  "name": "exampleAPI",  
  "version": "1"  
}
```



VERSIONING WITH ACCEPT HEADER

<http://bit.ly/troyhuntapi>

```
$ curl -H "accept: application/vnd.exampleapi.v2"  
      http://localhost:9000/version
```

```
{  
  "statusCode":200,  
  "applicationDetail":{  
    "name":"exampleAPI",  
    "version":"2"  
  }  
}
```



@DAVESTEVENS84

^ more stuff like this ^

dstevens@zappos.com

github.com/shakefon

github.com/shakefon/hapiday14