# Node:资源压缩之zlib

**概览**

做过web性能优化的同学，对性能优化大杀器**gzip**应该不陌生。浏览器向服务器发起资源请求，比如下载一个js文件，服务器先对资源进行压缩，再返回给浏览器，以此节省流量，加快访问速度。

浏览器通过HTTP请求头部里加上**Accept-Encoding**，告诉服务器，“你可以用gzip，或者defalte算法压缩资源”。

Accept-Encoding:gzip, deflate

那么，在nodejs里，是如何对资源进行压缩的呢？答案就是**Zlib**模块。

**入门实例：简单的压缩/解压缩**

**压缩的例子**

非常简单的几行代码，就完成了本地文件的gzip压缩。

var fs = require('fs');

var zlib = require('zlib');

var gzip = zlib.createGzip();

var inFile = fs.createReadStream('./extra/fileForCompress.txt')

var out = fs.createWriteStream('./extra/fileForCompress.txt.gz');

inFile.pipe(gzip).pipe(out);

**解压的例子**

同样非常简单，就是个反向操作。

var fs = require('fs');

var zlib = require('zlib');

var gunzip = zlib.createGunzip();

var inFile = fs.createReadStream('./extra/fileForCompress.txt.gz');

var outFile = fs.createWriteStream('./extra/fileForCompress1.txt'); inFile.pipe(gunzip).pipe(outFile);

**服务端gzip压缩**

代码超级简单。首先判断 是否包含 **accept-encoding** 首部，且值为**gzip**。

* 否：返回未压缩的文件。
* 是：返回gzip压缩后的文件。

var http = require('http');

var zlib = require('zlib');

var fs = require('fs');

var filepath = './extra/fileForGzip.html';

var server = http.createServer(function(req, res){ var acceptEncoding = req.headers['accept-encoding'];

var gzip; if(acceptEncoding.indexOf('gzip')!=-1){ // 判断是否需要gzip压缩 gzip = zlib.createGzip(); // 记得响应 Content-Encoding，告诉浏览器：文件被 gzip 压缩过 res.writeHead(200, { 'Content-Encoding': 'gzip' }); fs.createReadStream(filepath).pipe(gzip).pipe(res); }else{ fs.createReadStream(filepath).pipe(res); } }); server.listen('3000');

**服务端字符串gzip压缩**

代码跟前面例子大同小异。这里采用了**slib.gzipSync(str)**对字符串进行gzip压缩。

var http = require('http'); var zlib = require('zlib'); var responseText = 'hello world'; var server = http.createServer(function(req, res){ var acceptEncoding = req.headers['accept-encoding']; if(acceptEncoding.indexOf('gzip')!=-1){ res.writeHead(200, { 'content-encoding': 'gzip' }); res.end( zlib.gzipSync(responseText) ); }else{ res.end(responseText); } }); server.listen('3000');

**写在后面**

deflate压缩的使用也差不多，这里就不赘述。更多详细用法可参考[官方文档](https://nodejs.org/api/zlib.html" \l "zlib_class_options)。