02

OPEN ORIENTED

凹 凸 实 验 室

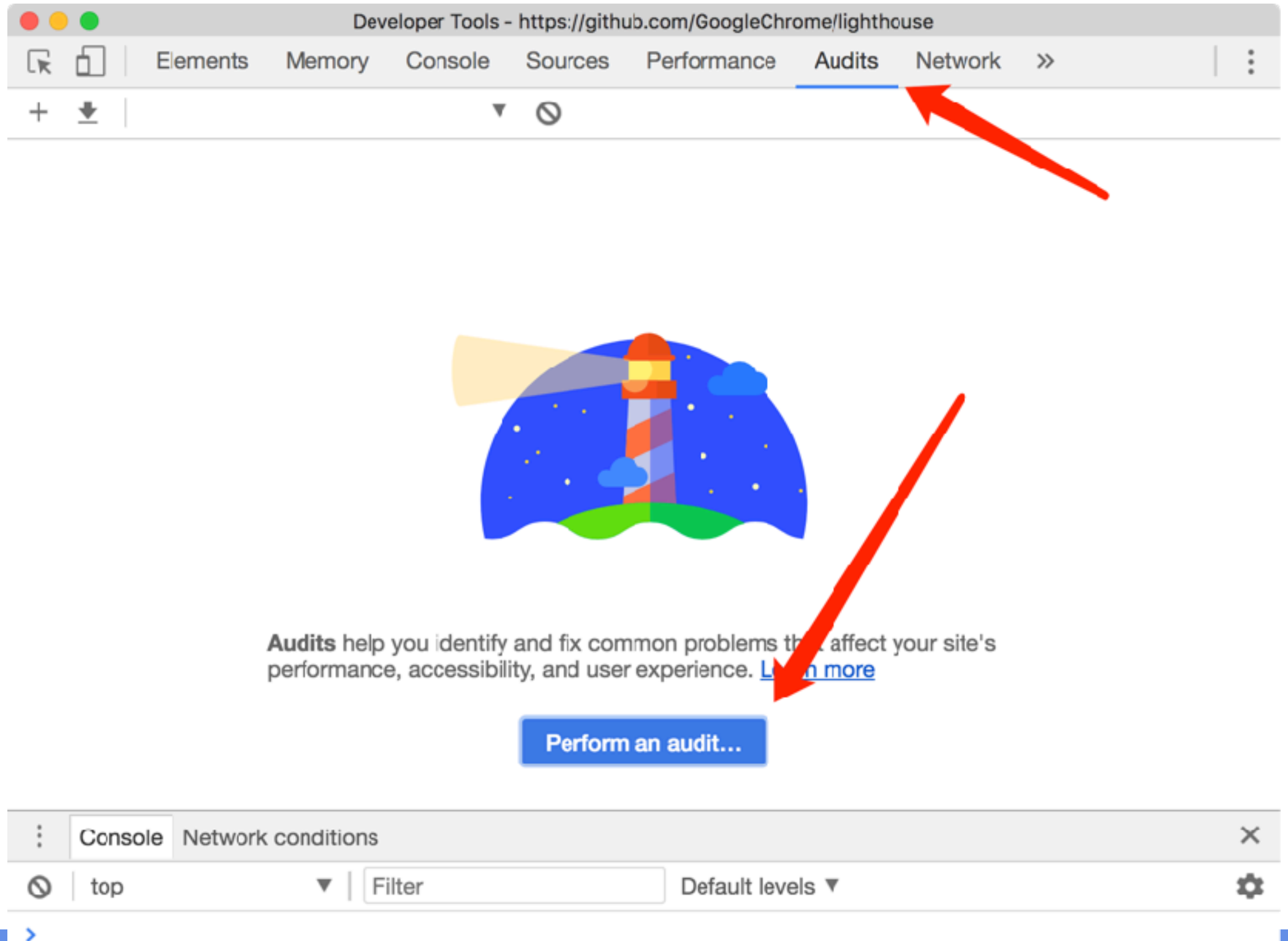
前端性能调优

余澈

- 没有原理分析
- 没有最佳实践
- 没有性能对比
- 没有温馨提示

当已经出现性能问题时，
如何去分析、定位并解决问题?

Lighthouse


Developer Tools - https://github.com/GoogleChrome/lighthouse
Elements
Memory
Console
Sources
Performance
Audits
Network
Audits help you identify and fix common problems that affect your site's performance, accessibility, and user experience. Learn more
Perform an audit...
Console
Network conditions
top
Filter
Default levels

Lighthouse CLI


```
yarn global add lighthouse

# or

npm install -g lighthouse
```

Lighthouse CLI


```
$ lighthouse --help

lighthouse <url>

Logging:
  --verbose  Displays verbose logging                                                                                    [boolean]
  --quiet    Displays no progress, debug logs or errors                                                                  [boolean]

Configuration:
  --save-assets                 Save the trace contents & screenshots to disk                                           [boolean]
  --save-artifacts              Save all gathered artifacts to disk                                                     [boolean]
  --list-all-audits             Prints a list of all available audits and exits                                         [boolean]
  --list-trace-categories       Prints a list of all required trace categories and exits                                [boolean]
  --additional-trace-categories Additional categories to capture with the trace (comma-delimited).
  --config-path                 The path to the config JSON.
  --chrome-flags                Custom flags to pass to Chrome (space-delimited). For a full list of flags, see
                                http://peter.sh/experiments/chromium-command-line-switches/.

                                Environment variables:
                                CHROME_PATH: Explicit path of intended Chrome binary. If set must point to an executable of a build of
                                Chromium version 54.0 or later. By default, any detected Chrome Canary or Chrome (stable) will be launched.
                                                                                                                [default: ""]
  --perf                        Use a performance-test-only configuration                                               [boolean]
  --port                        The port to use for the debugging protocol. Use 0 for a random port                  [default: 9222]
  --hostname                    The hostname to use for the debugging protocol.                                  [default: localhost]
  --max-wait-for-load           The timeout (in milliseconds) to wait before the page is considered done loading and the run should continue.
                                WARNING: Very high values can lead to large traces and instability                   [default: 25000]

Output:
  --output       Reporter for the results, supports multiple values                    [choices: "json", "html", "domhtml"] [default: "html"]
  --output-path  The file path to output the results. Use 'stdout' to write to stdout.
                 If using JSON output, default is stdout.
                 If using HTML output, default is a file in the working directory with a name based on the test URL and date.
                 If using multiple outputs, --output-path is ignored.
                 Example: --output-path=./lighthouse-results.html
  --view         Open HTML report in your browser                                                                        [boolean]

Options:
  --help                        Show help                                                                               [boolean]
  --version                     Show version number                                                                     [boolean]
  --blocked-url-patterns        Block any network requests to the specified URL patterns                                  [array]
  --disable-storage-reset       Disable clearing the browser cache and other storage APIs before a run                  [boolean]
  --disable-device-emulation    Disable Nexus 5X emulation                                                              [boolean]
  --disable-cpu-throttling      Disable CPU throttling                                                 [boolean] [default: false]
  --disable-network-throttling  Disable network throttling                                                              [boolean]
  --interactive                 Open Lighthouse in interactive mode                                                     [boolean]

Examples:
  lighthouse <url> --view                                              Opens the HTML report in a browser after the run completes
  lighthouse <url> --config-path=./myconfig.js                         Runs Lighthouse with your own configuration: custom audits, report
                                                                       generation, etc.
  lighthouse <url> --output=json --output-path=./report.json --save-assets  Save trace, screenshots, and named JSON report.
  lighthouse <url> --disable-device-emulation --disable-network-throttling  Disable device emulation
  lighthouse <url> --chrome-flags="--window-size=412,732"              Launch Chrome with a specific window size
  lighthouse <url> --quiet --chrome-flags="--headless"                 Launch Headless Chrome, turn off logging

For more information on Lighthouse, see https://developers.google.com/web/tools/lighthouse/.
```

Lighthouse CLI


```
lighthouse https://jd.com —disable-device-emulation
```

Lighthouse

Lighthouse
Version: 2.4.0

- Progressive Web App 36
- Performance 74
- Accessibility 89
- Best Practices 62

Results for: https://www.jd.com/
Sep 4, 2017, 10:20 AM CST · ▸ Runtime settings

36	74	89	62
Progressive Web App	Performance	Accessibility	Best Practices

Progressive Web App

These audits validate the aspects of a Progressive Web App, as specified by the baseline PWA Checklist.

36

8 Failed Audits

- Does not register a service worker ✗
- Does not respond with a 200 when offline ✗
  WARNING: You may be failing this check because your test URL (https://jd.com) was redirected to "https://www.jd.com/". Try testing the second URL directly.
- Does not use HTTPS: 1 insecure request found ✗
- Page load is fast enough on 3G ✓
  First Interactive was found at 7,210 ms, however, the network request latencies were not sufficiently realistic, so the performance measurements cannot be trusted.
- User will not be prompted to Install the Web App ✗
  Failures: No manifest was fetched, Site does not register a service worker, Service worker does not successfully serve the manifest's start_url.
- Is not configured for a custom splash screen ✗
  Failures: No manifest was fetched.

Chrome profile

DEMOS

jsPerf vs ESbench

- jsperf 有多个浏览器的测试结果
- ESbench 支持 ES6+
- ESbench UI 更现代
- 两者都基于 benchmark.js

git.io/hydraosx

编译阶段优化

Webpack vs Rollup

main.js (entry module)

```
1 import { cube } from './maths.js';
2 console.log( cube( 5 ) ); // 125
```

maths.js

```
1  // This function isn't used anywhere, so
2  // Rollup excludes it from the bundle...
3  export function square ( x ) {
4      return x * x;
5  }
6
7  // This function gets included
8  export function cube ( x ) {
9      // rewrite this as `square( x ) * x`
10     // and see what happens!
11     return x * x * x;
12 }
```

Webpack vs Rollup


```
/******/ (function(modules) { // webpackBootstrap
/******/  // The module cache
/******/  var installedModules = {};
/******/  // The require function
/******/  function __webpack_require__(moduleId) {
/******/    // Check if module is in cache
/******/    if(installedModules[moduleId])
/******/      return installedModules[moduleId].exports;
/******/    // Create a new module (and put it into the cache)
/******/    var module = installedModules[moduleId] = {
/******/      exports: {},
/******/      id: moduleId,
/******/      loaded: false
/******/    };
/******/    // Execute the module function
/******/    modules[moduleId].call(module.exports, module, module.exports, __webpack_require__);
/******/    // Flag the module as loaded
/******/    module.loaded = true;
/******/
/******/    // Return the exports of the module
/******/    return module.exports;
/******/  }
/******/  // expose the modules object (__webpack_modules__)
/******/  __webpack_require__.m = modules;
/******/  // expose the module cache
/******/  __webpack_require__.c = installedModules;
/******/  // __webpack_public_path__
/******/  __webpack_require__.p = "";
/******/  // Load entry module and return exports
/******/  return __webpack_require__(0);
/******/ })
/************************************************************************/
/******/ ([
/* 0 */
/*!*******************!*\
  !*** multi main ***!
  \*******************/
/***/ function(module, exports, __webpack_require__) {
```

```
1  'use strict';
2
3  // This function gets included
4  function cube ( x ) {
5      // rewrite this as `square( x ) * x`
6      // and see what happens!
7      return x * x * x;
8  }
9
10 console.log( cube( 5 ) ); // 125
```

Webpack vs Rollup

- Webpack 可以打包几乎任何资源
- Webpack 社区更大
- Rollup 专注于 ES2015 module
- Rollup 能大幅提升解析性能
- Webpack for apps, rollup for library

Webpack vs Rollup


```
/******/ (function(modules) { // webpackBootstrap
/******/  // The module cache
/******/  var installedModules = {};
/******/  // The require function
/******/  function __webpack_require__(moduleId) {
/******/    // Check if module is in cache
/******/    if(installedModules[moduleId])
/******/      return installedModules[moduleId].exports;
/******/    // Create a new module (and put it into the cache)
/******/    var module = installedModules[moduleId] = {
/******/      exports: {},
/******/      id: moduleId,
/******/      loaded: false
/******/    };
/******/    // Execute the module function
/******/    modules[moduleId].call(module.exports, module, module.exports, __webpack_require__);
/******/    // Flag the module as loaded
/******/    module.loaded = true;
/******/
/******/    // Return the exports of the module
/******/    return module.exports;
/******/  }
/******/  // expose the modules object (__webpack_modules__)
/******/  __webpack_require__.m = modules;
/******/  // expose the module cache
/******/  __webpack_require__.c = installedModules;
/******/  // __webpack_public_path__
/******/  __webpack_require__.p = "";
/******/  // Load entry module and return exports
/******/  return __webpack_require__(0);
/******/ })
/************************************************************************/
/******/ ([
/* 0 */
/*!*********************!*\
  !*** multi main ***!
  \*********************/
/***/ function(module, exports, __webpack_require__) {
```

Babel vs Bublé


```
// classes
class Circle extends Shape {
  constructor ( radius ) {
    super();
    this.radius = this;
  }

  area () {
    return Math.PI * Math.pow( this.radius, 2 );
  }
}

```

Babel vs Bublé


```
var _createClass = function () { function defineProperties(target, 

function _classCallCheck(instance, Constructor) { if (!(instance in

function _possibleConstructorReturn(self, call) { if (!self) { thro

function _inherits(subClass, superClass) { if (typeof superClass !=

var Circle = function (_Shape) {
  _inherits(Circle, _Shape);

  function Circle(radius) {
    _classCallCheck(this, Circle);

    var _this = _possibleConstructorReturn(this, (Circle.__proto__

    _this.radius = _this;
    return _this;
  }

  _createClass(Circle, [{
    key: "area",
    value: function area() {
      return Math.PI * Math.pow(this.radius, 2);
    }
  }]);

  return Circle;
}(Shape);
```

```
// classes
var Circle = (function (Shape) {
  function Circle ( radius ) {
    Shape.call(this);
    this.radius = this;
  }

  if ( Shape ) Circle.__proto__ = Shape;
  Circle.prototype = Object.create( Shape && Shape.prototype );
  Circle.prototype.constructor = Circle;

  Circle.prototype.area = function area () {
    return Math.PI * Math.pow( this.radius, 2 );
  };

  return Circle;
}(Shape));
```

classes tests	**babel**	1.9x slower	9x slower
	babel-loose	1.7x slower	9x slower
	babel-runtime	1.8x slower	9x slower
	buble	Identical	Identical
	traceur	Identical	Identical
	typescript	Identical	Identical
	es6	Identical	Identical

Babel vs Bublé

- Babel 是模块化的设计，功能由插件完成
- Babel 支持更多的语法，也能针对某一语法进行单独优化
- Bublé 专注于打包 ES6
- Bublé 性能更好
- Bublé 托管在 gitlabs

THANKS

FOR YOUR WATCHING

02

OPEN ORIENTED

凹 凸 实 验 室