WHAT ARE WE DOING?

- Improve mobile web site
- Responsive future
- New www via the back door?
HOW ARE WE DOING IT?

- Builds on Guardian APIs
  - http://www.guardian.co.uk/open-platform
- Amazon hosted
  - EC2, S3, Cloudformation
- Code in public Github repo
  - http://github.com/guardian/frontend
**AGILE?** (or something)

- **Metrics and monitoring**
  - Omniture, Ganglia, Graphite, custom tools
- **Automation**
  - Tests, build, sprites, cloudformation
- **Continuous deployment**
  - merge to master, Team City, test, deploy
var guardian = {
    isModernBrowser: (
        'querySelector' in document &&
        'addEventListener' in window &&
        'localStorage' in window
    )
};

Friday, 18 January 13
MICRO LIBRARIES

- EventEmitter
- request
- bonzo
- qwery
- bean
```javascript
var r = new Router();

// Init all common modules first
Common.init(config);

r.get('/sport', function(req) {
    SportFront.init(req, config);
});

r.get('/football/:action/:day', function(req) {
    Football.init(req, config);
});
```

Friday, 18 January 13
REQUIREJS OPTIMIZER

- Uglify
- One app file
- Splitting up app file
- Bower
AMD

define(['common', 'modules/detect', 'bonzo'],
    function (common, detect, bonzo) {
        var theModule = {
            method: function () {
                if (detect.touch) {
                    common.doSomething();
                }
            }
        }
    }
);

return theModule;
CURLJS

- Asynchronous AMD loader
- 2.5Kb compressed
- Supports non AMD via plugin
- (Less jQuery compatible)
BOWER

> bower update bonzo

REQUIREJS PATHS

paths: {
  "bower": "components/bonzo/bonzo.min.js",
}

Friday, 18 January 13
BOWER

> bower update clearless

LESS BUILD

@import "components/clearless/mixins/helpers.less"
SMACSS

- base, layout, module, state
- optimise for change
- not ‘pretend semantics’
.pull-left {
  float: left;
}

.margin-bottom {
  margin-bottom: @baseline*2;
}

.gutter-right {
  padding-right: @gridGutter;
}

.border-top {
  border-top: 1px solid #BEBEBE;
}
STYLE GUIDE

- Ensures you’ve built a system
- Modules decoupled
- Type decoupled
LESS

- CSS pre-processor
- Mainly use variables and mixins
- No nesting (except @media)
NODEJS
BUILD PROCESS

- sbt calls out to nodejs
- JavaScript: requirejs, jshint
- CSS: Less (CSS lint?)
- Spriting: SVG & PNG
- Fonts: base64 encoded WOFF
ICON SPRITES

- Receive SVGs from design team
- We decide on format
- grunticon from Filament group
- Outputs CSS with SVG data URLs
- Outputs CSS / sprited PNG file
WEB FONTS

- Only support WOFF (nearly)
- 5(!) fonts totalling 97Kb
- Base64 encoded
- Cached in localStorage
THE USUAL STUFF

- Everything compressed
- Static assets with 10 year expiry
- One CSS file (so far?)
- One JS file (so far?)
CUTTING THE MUSTARD

- No JavaScript
- All low res images
- No web fonts
RESPONSIVE IMAGES?

- Not responding to viewport
- Responding to size of image
- Responding to network speed (sort of)
```
var loadTime = getPageSpeed();
var speed = 'low';

if (loadTime < 1000) {
    speed = 'high';
}

return speed;
```
MONITORING SPEED
OTHER TOOLS

- WebPagetest
- Google Critical Path Explorer
- PhantomJS
THIRD PARTIES?

- Removing SPOFs
- document.write()
- Ads
Thank-you!

http://lanyrd.com/ckwqy
@andyhume