Project Management and Automation
Using Maven and Grunt to accelerate development
Matthew Hanlon • SEA Conference • April 7, 2014
Introduction
Web and Mobile Applications
Building tools and applications to make scientists more productive.

7 Developers (and hiring!)
10+ projects
Without process
Boring
What’s the big deal?
Too many results in published scientific papers are not reproducible

- The data is unavailable
- The environment no longer exists
- The code is broken!
Reproducibility in Computer Science

That's only a 20% success rate!

[source: http://reproducibility.cs.arizona.edu/]
Why doesn’t the code build?

- Missing dependencies
- Mysterious configuration parameters
- Compile errors
- And of course, there is no documentation
How can we fix that?

Automate the boring stuff

Make the rest worth it
Tools for automation and process

- Apache Maven
- Grunt.js
- Bower
- Sass/Compass
- Doxygen
Maven
Managing project build, reporting, and documentation.
<table>
<thead>
<tr>
<th>Category</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project information</td>
<td>Name, version, URL, developers, contributors, licensing, organization</td>
</tr>
<tr>
<td>Build settings</td>
<td>Build process, dependencies, non-code resource handling, reporting</td>
</tr>
<tr>
<td>Environment settings</td>
<td>Source control, repositories, distribution, issue management</td>
</tr>
</tbody>
</table>
Plugins

- Basics: Compiler, Resources, JavaDocs, Eclipse, IntelliJ IDEA
- Advanced: Exec, Site
- Community: Doxygen, Liferay, Jetty
<project>
  <modelVersion>4.0.0</modelVersion>
  <groupId>com.github.mrhanlon</groupId>
  <artifactId>sea2014-demo</artifactId>
  <version>1.0-SNAPSHOT</version>
  <packaging>war</packaging>
  <dependencies>
    <dependency>
      <groupId>org.springframework</groupId>
      <artifactId>spring-context</artifactId>
      <version>${spring.version}</version>
    </dependency>
    <dependency>
      <groupId>org.springframework</groupId>
      <artifactId>spring-webmvc</artifactId>
      <version>${spring.version}</version>
    </dependency>
    <dependency>
      <groupId>org.springframework</groupId>
      <artifactId>spring-orm</artifactId>
      <version>${spring.version}</version>
    </dependency>
  </dependencies>
</project>
$ mvn install
[INFO] Scanning for projects...
[INFO] Using the builder org.apache.maven.lifecycle.internal.builder.singlethreaded.SingleThreadedBuilder
[INFO] ------------------------------------------------------------------------
[INFO] Building module1 1.0-SNAPSHOT
[INFO] ------------------------------------------------------------------------
[INFO] --- maven-clean-plugin:2.5:clean (default-clean) @ module1 ---
[INFO] --- maven-resources-plugin:2.6:resources (default-resources) @ module1 ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] skip non existing resourceDirectory /Users/mrhanlon/workspace/github/mrhanlon/sea2014-demo/module1
[INFO] --- maven-compiler-plugin:2.0.2:compile (default-compile) @ module1 ---
[INFO] Compiling 1 source file to /Users/mrhanlon/workspace/github/mrhanlon/sea2014-demo/module1
[INFO] --- maven-resources-plugin:2.6:testResources (default-testResources) @ module1 ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] skip non existing resourceDirectory /Users/mrhanlon/workspace/github/mrhanlon/sea2014-demo/module1
Grunt
The JavaScript Task Runner
(It's not just for JavaScript!)
Gruntfile.js
Declarative JSON/JavaScript configuration file
Executes tasks in Node.js runtime
Loads of community plugins:

- jslint/jshint
- file tasks
- watch
- compass
- library/framework support
'use strict';

module.exports = function(grunt) {

    // Project configuration.
    grunt.initConfig({
        // Metadata.
        pkg: grunt.file.readJSON('tiny-pubsub.jquery.json'),
        banner: '/*! <%= pkg.title %><%= pkg.name %> - v<%= pkg.version %> - ' +
              '<%= grunt.template.today("yyyy-mm-dd") %> */
               '*/
        ,
        // Task configuration.
        clean: {
            src: ['dist']
        },
        concat: {
            options: {
                banner: '<%= banner %>',
                stripBanners: true
            }
        },
        uglify: {
            options: {
                banner: '<%= banner %>',
                stripBanners: true
            }
        },
        qunit: {
            files: ['test/**/*\.html']
        },
        jshint: {
            options: {
                banner: '<%= banner %>',
                stripBanners: true
            }
        }
    });
$ grunt
Running "jshint:gruntfile" (jshint) task
>> 1 file lint free.

Running "jshint:src" (jshint) task
>> 1 file lint free.

Running "jshint:test" (jshint) task
>> 1 file lint free.

Running "qunit:files" (qunit) task
Testing test/tiny-pubsub.html....OK
>> 4 assertions passed (23ms)

Running "clean:files" (clean) task
Cleaning "dist"...OK

Running "concat:dist" (concat) task
File "dist/ba-tiny-pubsub.js" created.

Running "uglify:dist" (uglify) task
File "dist/ba-tiny-pubsub.min.js" created.
Bower

Front-end package management

package agnostic

runs over git
Declarative JSON configuration

```json
{
    "name": "vislab-reservation-portlet",
    "version": "1.0.0",
    "dependencies": {
        "modernizr": "~2.6.2",
        "jqueryui-timepicker-addon": "1.4.3",
        "mustache": "~0.8.1",
        "datejs": "*",
        "fullcalendar": "~1.6.4"
    }
}
```
Sass/Compass

"Expressive" CSS
@mixin box-sizing($bs) {
  $bs: unquote($bs);
  @include experimental(box-sizing, $bs,
    -moz, -webkit, not -o, not -ms, not -khtml, official
  );
} *

* {
  @include box-sizing(border-box);
}
.alert {
  border: 1px solid black;
}
.alert-error {
  @extend .alert;
  color: red;
  border-color: red;
}
Doxygen
Generating documentation for annotated sources
Tons of language support
Extensions for other languages
(like JavaScript!)
Demo
So what’s next?
Get up and running in minutes

```bash
$> git clone <repo url> my-dev-env
$> cd my-dev-env
$> git submodule init
$> git submodule update
$> mvn install
$> mvn liferay:deploy
```
YOU MUST CHOOSE

BUT CHOOSE WISELY
The Second Law of Thermodynamics

The entropy of an isolated system never decreases, because isolated systems always evolve toward thermodynamic equilibrium a state with maximum entropy.
Peep behind the scenes of Chaos Software Ltd. to discover how it keeps their software services under control.