

前端工程化在陆金所的 实践

陆金所 @见见

too young
too simple?

项目中常见处理

- * 框架/库选型
- * 初始化项目
- * 优化构建流程
- * 模块化开发

项目中常见处理

- * **js** 模块依赖管理
- * 语法转义 (**babel**)
- * **css** 预处理
- * 优化（合并，压缩）
- * ...

为什么工程化

* 解放劳动力

90s 原则

* 实现个体的最大化

一个项目，分配到一群人
处理，还是能像一个人一
样无缝合作

* 不再造轮子

前端工程的本质

- * 便捷性

- * 规范性

- * 复用性

在陆金所的实践

脚手架

```
$ yo @lu/luffy
```

```
  _-----_
  |         |
  |--(o)--|
  \-----/
  ( _`U`_ )
  /___A___\  /
    |   ~   |
    '       '
  _.-'-. _.-'-. _.-'-.
  |  °  | °  | °  | °  | °
  '-----'-----'-----
```

```
Welcome to the finest
generator-luffy
generator!
```

```
? Your project name lu
```

```
? Which version is selected? (Use arrow keys)
```

```
> react
```

```
ie8
```

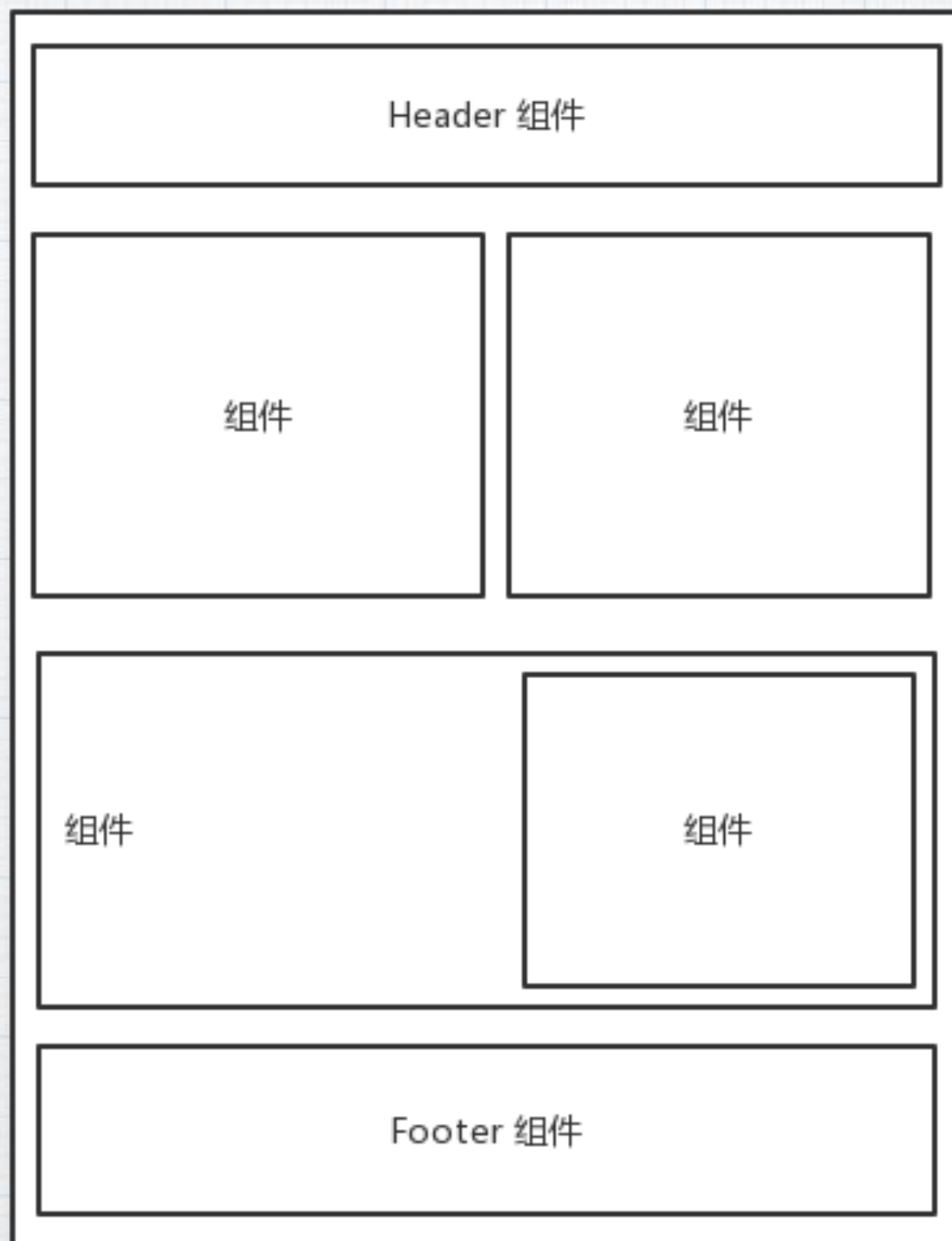
项目结构



组件化开发

- * 基础组件 **luvi, lubase**
- * 共用业务组件 **lubc**
- * 项目组件化

组件化开发



页面
Page

项目构建 **lulu-cli**

- * 集中管理 **webpack** 的 **loader plugin** 以及 **babel** 等相关配置
- * 自定义扩展配置文件

\$lulu init

\$? Dev Server Port? 8080

? What is the entry file path? ./src/index.js

\$? Give the entry a name? main

\$ create : lulu.config.js

\$run lulu start to begin

\$lulu start

\$lulu build

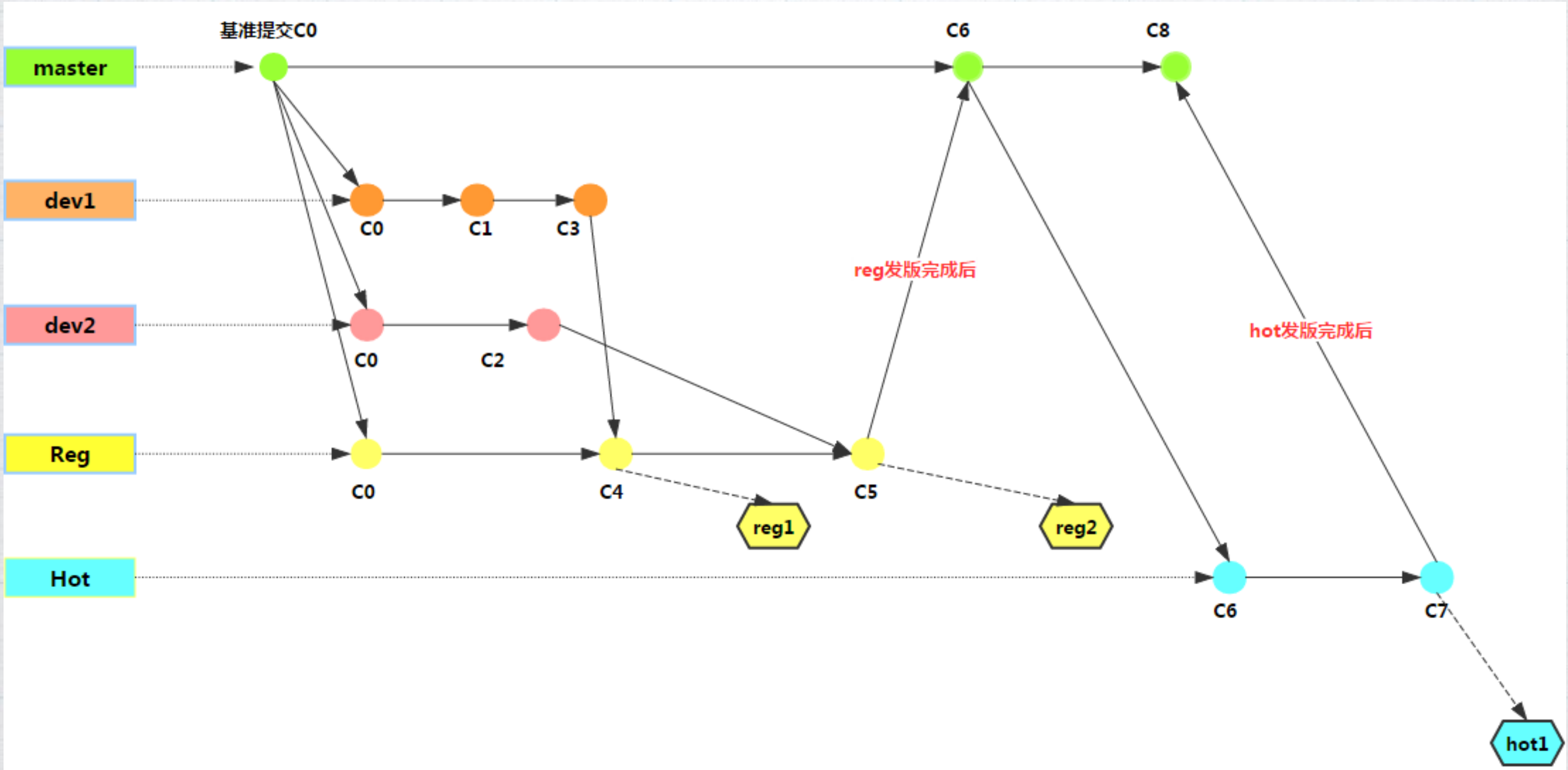
lulu.config.js

```
module.exports = {
  preset: 'ie8',
  webpack: function (config) {
    console.log(entryList)
    config.entry = entryList
    config.resolve.alias = {
      'COMMON': path.resolve(__dirname, `${STATIC_PATH}/common`)
    }
    config.output.path = path.join(__dirname, './dist')
    return config
  },
  devServer: {
    port: 3123,
    hot: true,
    setup: function (app) {
      app.use('/secmkt', express.static(path.resolve(__dirname, STATIC_PATH)))
    },
    publicPath: 'http://127.0.0.1:3123/secmkt'
  }
}
```

发布与部署

git 分支策略

- * 对应 **Reg/Quick/Hot/Emp** 分支
- * 对应的 **gitflow** 操作



gulpfile.js

```
+ gulp.task('min-script', function () { ...  
  })  
  
+ gulp.task('min-css', function () { ...  
  })  
  
+ gulp.task(['min-image', function () { ...  
  }])  
  
+ gulp.task('rev', ['min-script', 'min-css', 'min-image'], function () { ...  
  })  
  
+ gulp.task('packageStatic', ['rev'], function () { ...  
  })
```

提供可选 task npm包

- ▲ @lu

- ▲ web-resource-optimizer

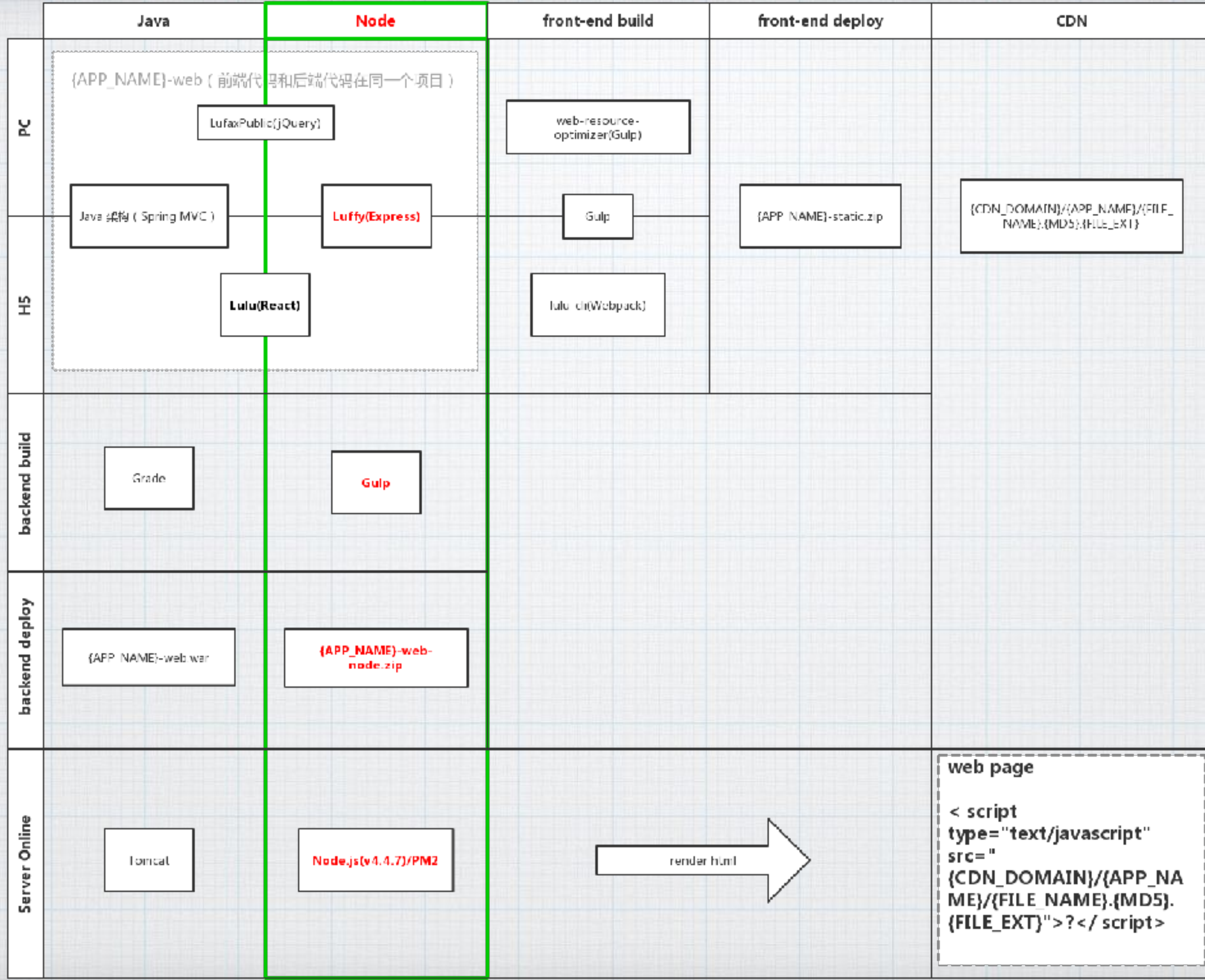
- ▲ lib

- JS build.js

- JS clean.js

- JS jshint.js

- JS rev.js



未来备选方案

- * Dock 部署

- * Serverless 化

- * ...

THX.