



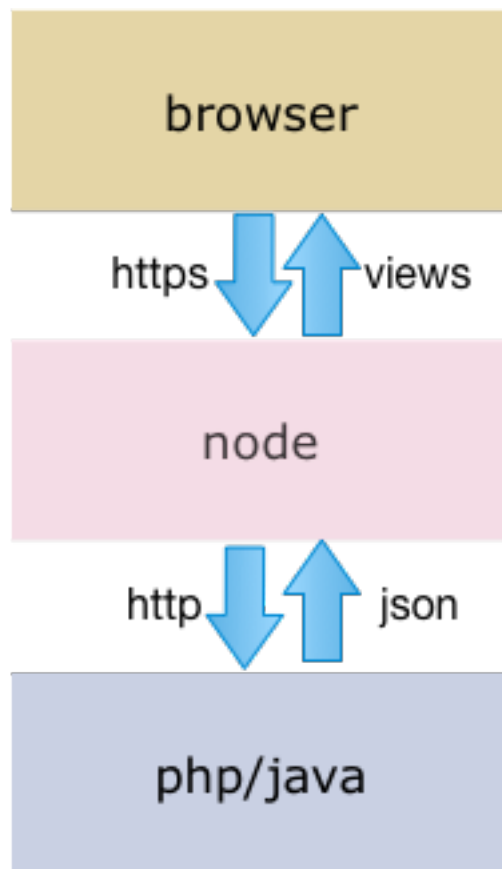
前后端分离后的线上稳定性监控

分享人：韩路

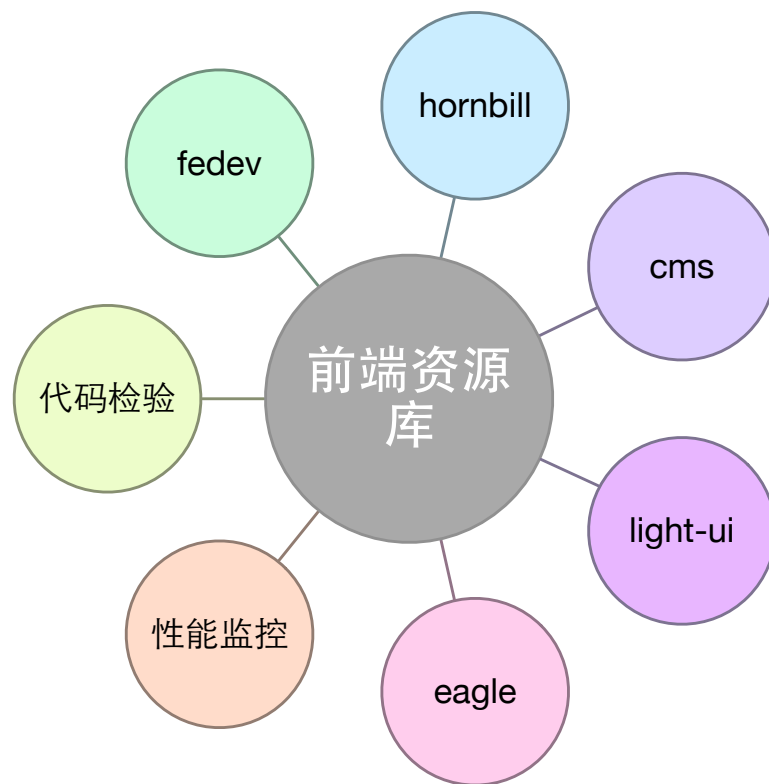
要讲的内容

- 1、eagle是什么？
- 2、eagle的实现？
- 3、eagle还面临哪些问题？

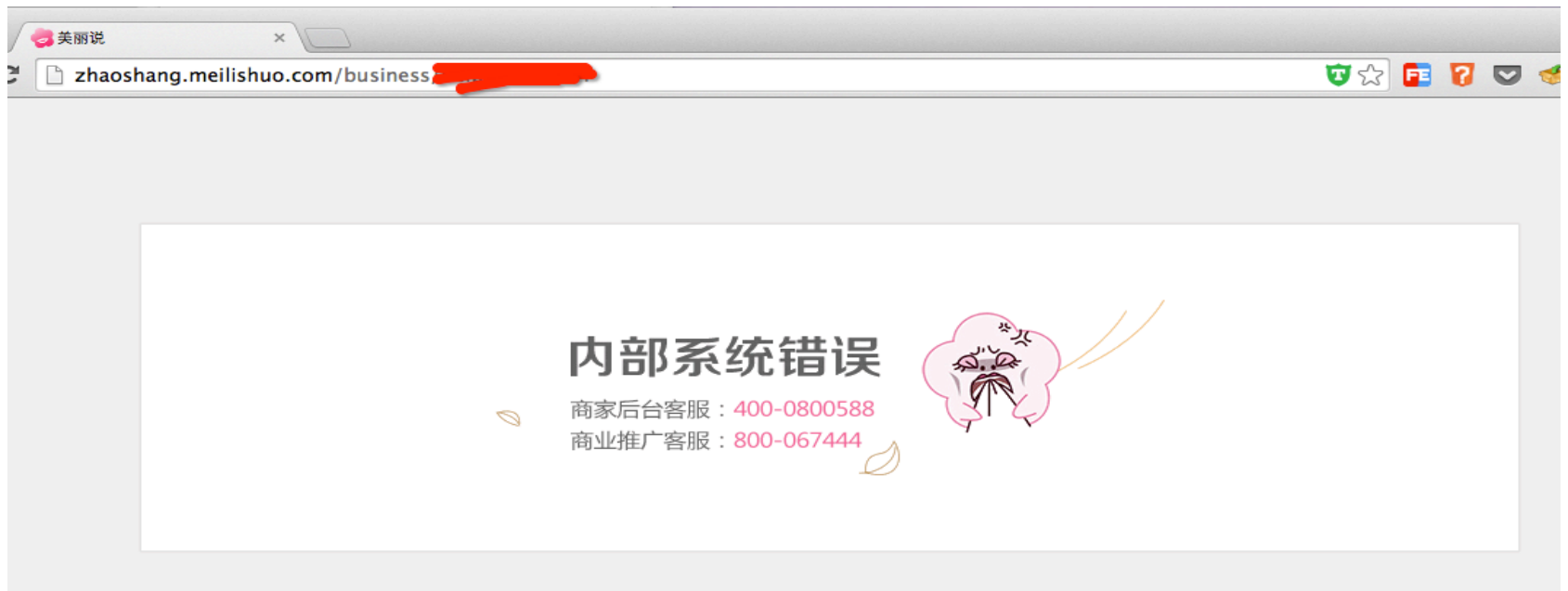
Eagle背景



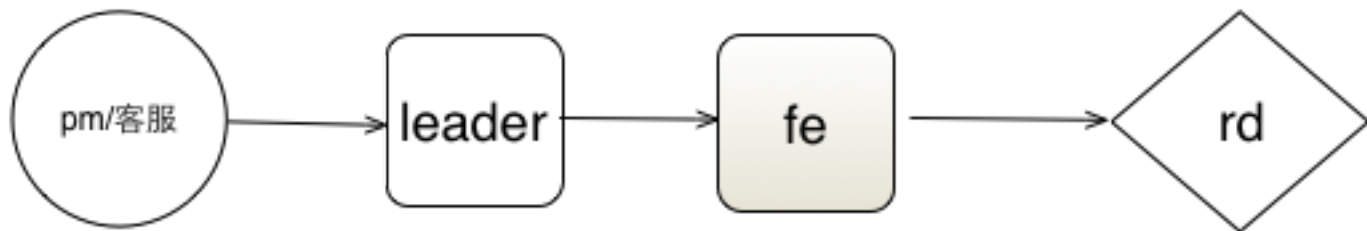
商业前端资源库



线上500



他们是这样解决问题的



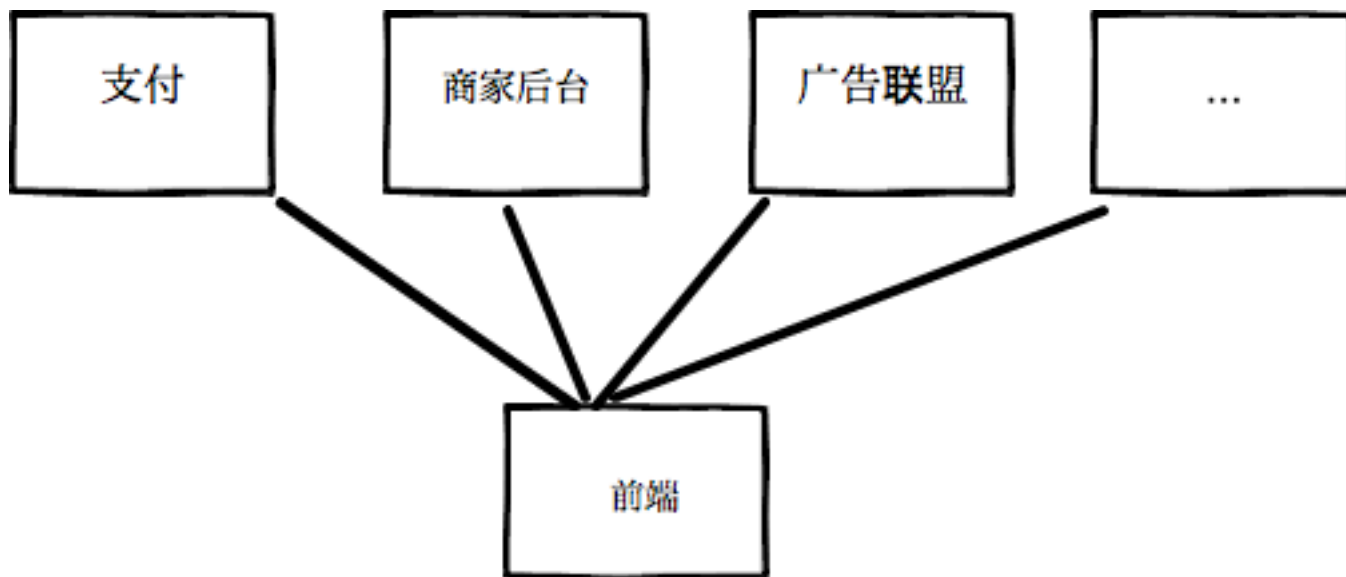
为什么要用eagle?

- 1、沟通成本
- 2、靠常用手段（例如：查看错误栈、复现、查看数据等等）
- 3、人力分配问题（究竟谁去解这个bug）

什么是Eagle ?

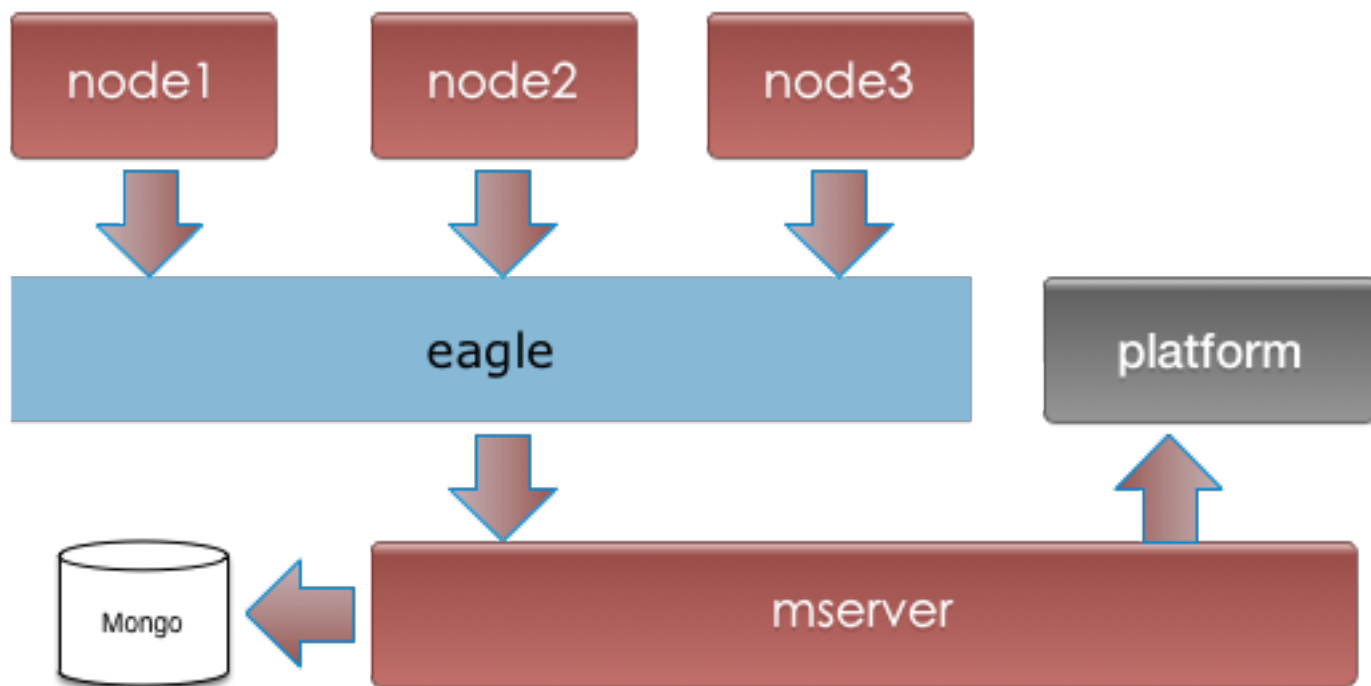


为什么前端做？

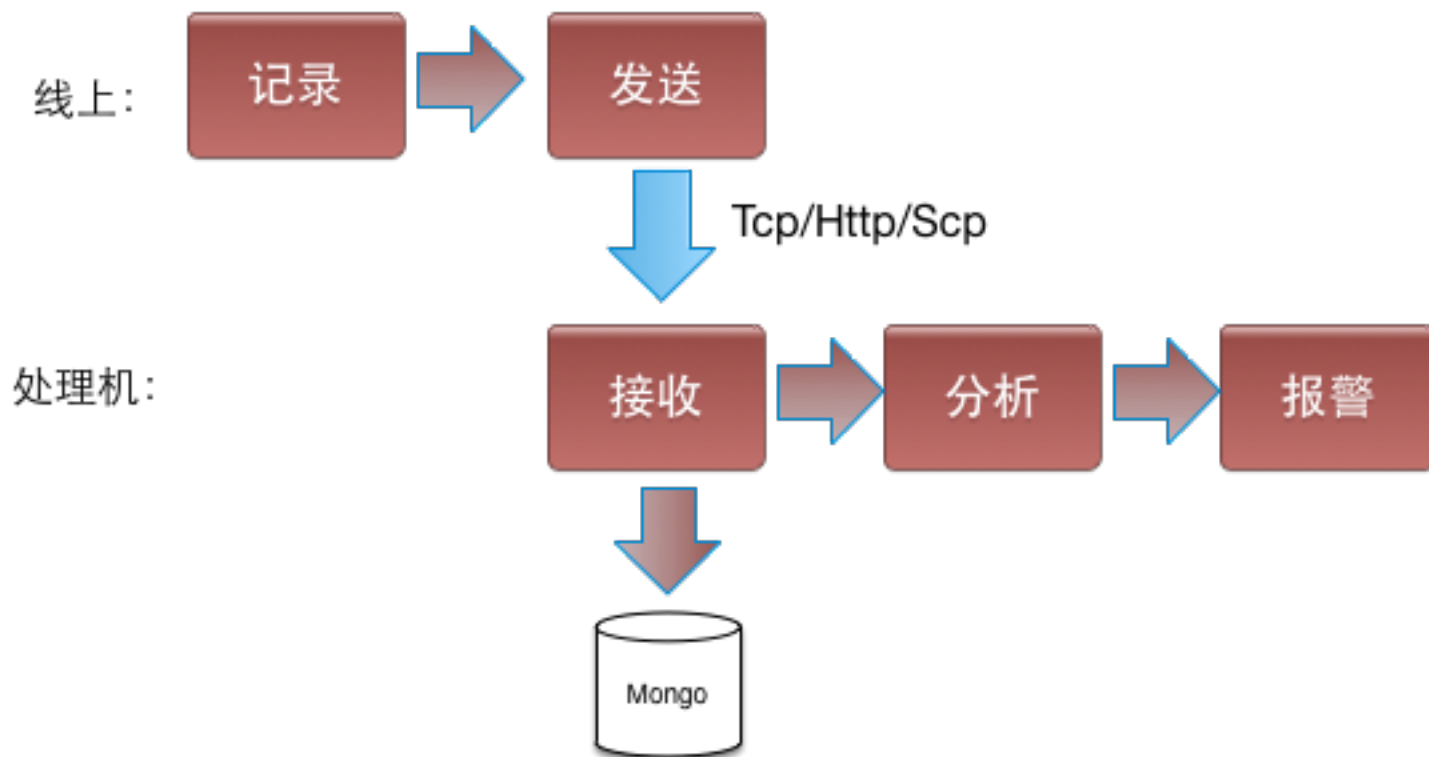


Eagle实现

Eagle数据流向



处理日志



一般会有哪些错

- 1、fatal (node模板)
- 2、brwoser (浏览器)
- 3、api (后端错误)

模板错误

```
try {  
  makeCaughtError();  
  res.end('ok');  
} catch (e) {  
  console.log(e.stack);  
  res.end('error');  
}
```

```
process.on('uncaughtException', function(err) {  
  console.dir(err);  
})
```

错误解析

```
function detectLog(err, req) {  
  var stack = err.stack.split('\n'),  
      errDetail = {};  
  errDetail = {  
    msg: stack[0],  
    file: (/at\s.+\/\//.test(stack[1]) ? stack[1] : stack[2]).match(/\(([^\)]*)\)/)[1],  
    host: req.headers.host,  
    url: req.url  
    ...  
  }  
  return errDetail;  
}
```

浏览器记录

```
<script type="text/javascript">
    window.onerror = function(sMessage,sUrl,sLine){
        console.dir(arguments);
        new Image().src = 'http://xxx.xxx.xxx'
    };
</script>


<script>
    /*比较重要的业务模块*/
    function business(){
        try{
            JSON.parse('');
        }catch(e){
            var fun = 'business',
                filename = 'test',
                stack = e.stack;
            ...
        }
    }
</script>
```


浏览器多元化

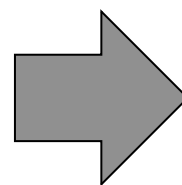


后端接口数据

```
{
  "code": 0,
  "data": {
    "info": [
      { ... }, // 7 items
      { ... } // 7 items
    ],
    "meta": { ... } // 6 items
  }
}
```



```
▼ {  
  "code": 0,  
  "data": {  
    "info": [  
      ▶ { ... }, // 7 items  
      ▶ { ... } // 7 items  
    ],  
    "meta": { ... } // 6 items  
  }  
}
```



```
▼ {  
  "code": 0,  
  "data": {  
    "info": []  
  }  
}
```

api错误记录

```
function detectApi(remote, req) {  
  var result = remote.result;  
  if (result.hasOwnProperty('code')) {  
    if (result.code == 0 && result.hasOwnProperty('info') && typ  
      return true;  
    }  
  }  
  return {  
    url: req.url,  
    msg: {  
      api: remote.remoteUri,  
      status: remote.status,  
      result: result  
    }  
    ...  
  }  
}
```

记录日志

```
{  
  "level": "api",  
  "ip": "202.107.226.86",  
  "url": "zhaoshang.meilishuo.com/business/help?cid=107",  
  "resTime": "18ms",  
  "referer": "http://zhaoshang.meilishuo.com/business/help?cid=151",  
  "userAgent": "Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.1; SV1; .NET CLR 1.1.4",  
  "time": "2015-03-25 11:47:12",  
  "status": 200,  
  "msg": "{\\"api\\":\\"/brdwiki/detail?cid=107\\",\\"status\\":200,\\"useTime\\":15,\\"result\\":{\\"\\\\"error_code\\"\\":22220}\\"}",  
  "php": null,  
  "user_id": "",  
  "shop_id": "",  
  "host_name": "dfz-bizfe-03"  
}
```

线上log发送

- 1、周期scp
- 2、实时发送内容

Tail+TCP

```
this.tail = child_process.spawn('tail', logFiles);
this.tail.stdout.on('data', function (data) {
  if (!isFirstConnect) {
    me.buffer.push(data.toString('utf-8'));
  }
  if (reg.test(data.toString('utf-8'))) {
    isFirstConnect = false;
  }
});
setInterval(function () {
  me.client.write(me.buffer.shift(), 'utf8');
}, 100);
```

可能要注意的

```
4. bash
hirra:~ a2014$ ps -efl grep tail
  501   721   611    0 12:21上午 ttys002    0:00.07 node tail.js
  501   722   721    0 12:21上午 ttys002    0:00.01 tail -f /tmp/test.js
  501   727   696    0 12:21上午 ttys004    0:00.00 grep tail
hirra:~ a2014$ kill 721
hirra:~ a2014$ ps -efl grep tail
  501   722     1    0 12:21上午 ttys002    0:00.01 tail -f /tmp/test.js
  501   730   696    0 12:22上午 ttys004    0:00.00 grep tail
hirra:~ a2014$
```


处理机目录结构

```
▼ eagle
  ► cmd
  ► crontab
  ► libs
  ► models
  ► node_modules
  ▼ service
    ▼ analyze
      ► filter
        analyze.js
      ► notify
      ► sink
    ► test
      config.js
      eagle.js
      package.json
```

报警

- 1、Case by case
- 2、找到指定人
- 3、邮件、短信各种骚扰渠道

策略

```
strategy: {  
  fatal: {  
    num: 300, //多少个数的的时候会报  
    period: 60, //周期  
    interval: 180, //间隔多少秒 再报  
    instantInterval: 120 //实时的fatal错误, 邮件间隔120秒  
  },  
  api: {  
    apiLists: [  
      '/shopadmin/id_info',  
      '/shopadmin/menu',  
      ...  
    ],  
    importantNum: 500, //重要的短信阈值  
    num: 500, //多少个数的的时候会报  
    period: 60, //周期  
    interval: 180 //间隔多少秒 再报  
  }  
}
```

如何指派人

- 1、指定模块负责人
- 2、svn log、info

Svn log

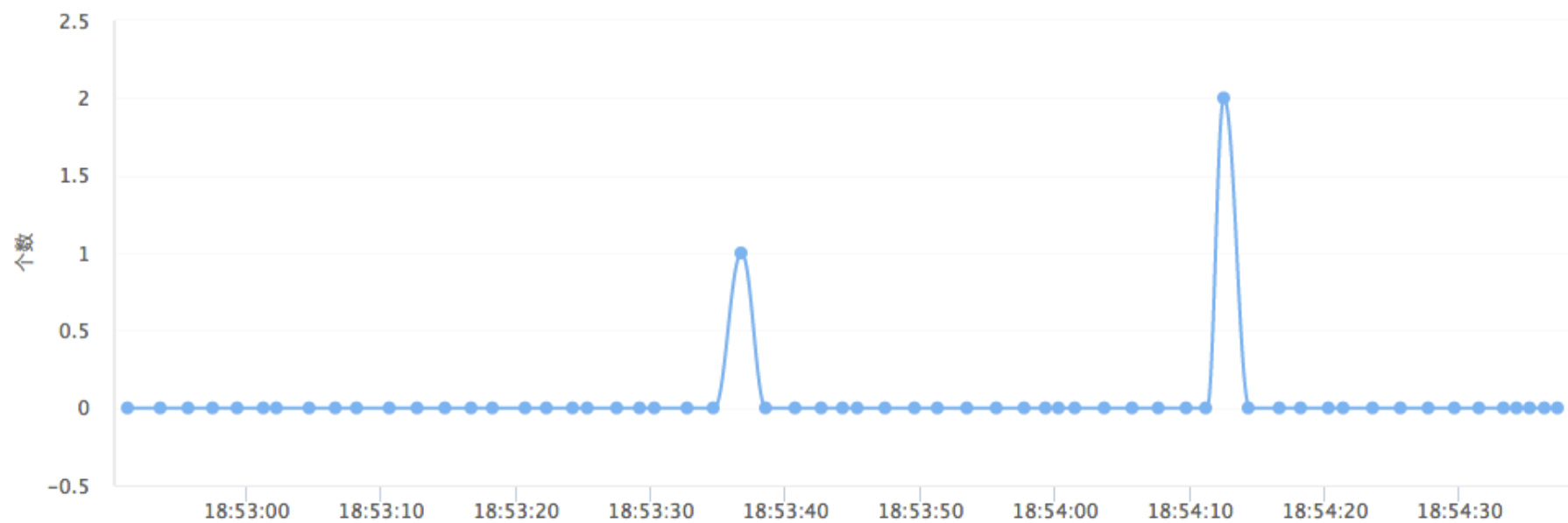
```
1. bash
hirra:dc a2014$ svn log flow
-----
r301237 | luhan | 2015-03-15 20:07:06 +0800 (日, 15 3 2015) | 1 line
增加实时数据、修改单品、样式, qa可忽略 hirra
-----
r301204 | luhan | 2015-03-15 15:13:15 +0800 (日, 15 3 2015) | 1 line
修改数据中心样式、bootstrap.css; 增加流量分析、实时数据 hirra
-----
```

Svn info

```
1. bash
hirra:dc a2014$ svn info flow
Path: flow
Working Copy Root Path: /Users/a2014/projects/biz/shop
URL: http://svn.meilishuo.com/repos/meilishuo/fex/biz/trunk/shop/views/dc/flow
Repository Root: http://svn.meilishuo.com/repos
Repository UUID: 66a6744b-ee30-477d-922d-9bb399cb4c38
Revision: 310284
Node Kind: directory
Schedule: normal
Last Changed Author: luhan
Last Changed Rev: 301237
Last Changed Date: 2015-03-15 20:07:06 +0800 (日, 15 3 2015)
```

Node拿到Svn log

```
child_process.exec('svn info ' + path, function (err, data) {  
  if(!err){  
    //对data做些处理  
    var eachLine = data.split('\n');  
    ...  
    callback && callback(err,result);  
  }  
});
```



内容管理系统

url id

被指派

| 级别 | 模块 | 路径 | 时间 | 出现次数↓ | 指派 | 处理 | 详情 |
|-------|------|--------------------------------|---------------------|-------|----------------------------------|----|----|
| fatal | shop | shop.meilishuo.com/deal/order/ | 2015-03-13 18:32:01 | 1068 | <input type="text" value="请选择"/> | | 详情 |
| fatal | shop | shop.meilishuo.com/home/ | 2015-03-13 18:31:49 | 989 | <input type="text" value="请选择"/> | | 详情 |

错误日志报告

刚用eagle的时候

--- bizfe 发给

检测结果: 共0个warning, 0个error,782个fatal,39579个api

用了eagle三个月后...

--- bizfe 发给

检测结果: 共0个warning, 0个error,4个fatal,4505个api

日志详情

| | | | | | |
|-----------|---|---------|---------------------------------------|---|--------------|
| 模块 | shop | URL | shop.meilishuo.com/deal/order/ 模拟登录>> | | |
| 时间 | 2015-03-13 18:32:01 | 用户ID | | 店铺ID | |
| 用户所在地 | | 错误级别 | fatal | 前端机器 | dfz-bizfe-06 |
| UserAgent | Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/37.0.2062.120 Safari/537.36 | | | | |
| 出错文件 | shop/views/common/nav.html (第94行, 第15列) | | | | |
| 错误源代码 | Meilishuo.constant.current_url = '<%= urls.join("") %>'; | | | | |
| 错误信息 | __htm += urls.--->>>join(""); at exports.html (/home/meilishuo/biz/hornbill/tmp/shop93e4684291a2a4ec1e32a183b668f4cc.est:61:15) TypeError: Cannot call method 'join' of undefined | | | | |
| 操作记录 | 时间 | 下达人 | 操作 | 执行人 | 留下的话 |
| PHP接口请求 | 接口名称 | HTTP状态码 | 解释 | 返回数据 | |
| | shop::security/needverify | 200 | 请求成功 | {"code":0,"message":"不需要验证","info":{"needverify":0,"status":"0"}} | |

短信

3-14 9:37

1分钟内出现了500个PHP接口报错，其中md出现了1次，shop出现了499次。其中TOP3为：接口shop::/shopadmin/shop_healthy_new出现了447次，接口shop::/security/needverify出现了33次，接口shop::/user/log_on出现了15次。【美丽说】

邮件

2015-3-25前端线上【Node层】bug【fatal】

bizfe 发给 fe-biz, store-kernel, luhan

60秒内，Node层出现了以下bug:

zhaoshang模块:

url:zhaoshang.meilishuo.com/business,

还请前端@韩路同学跟进一下。

详情:[点击进入详情](#)

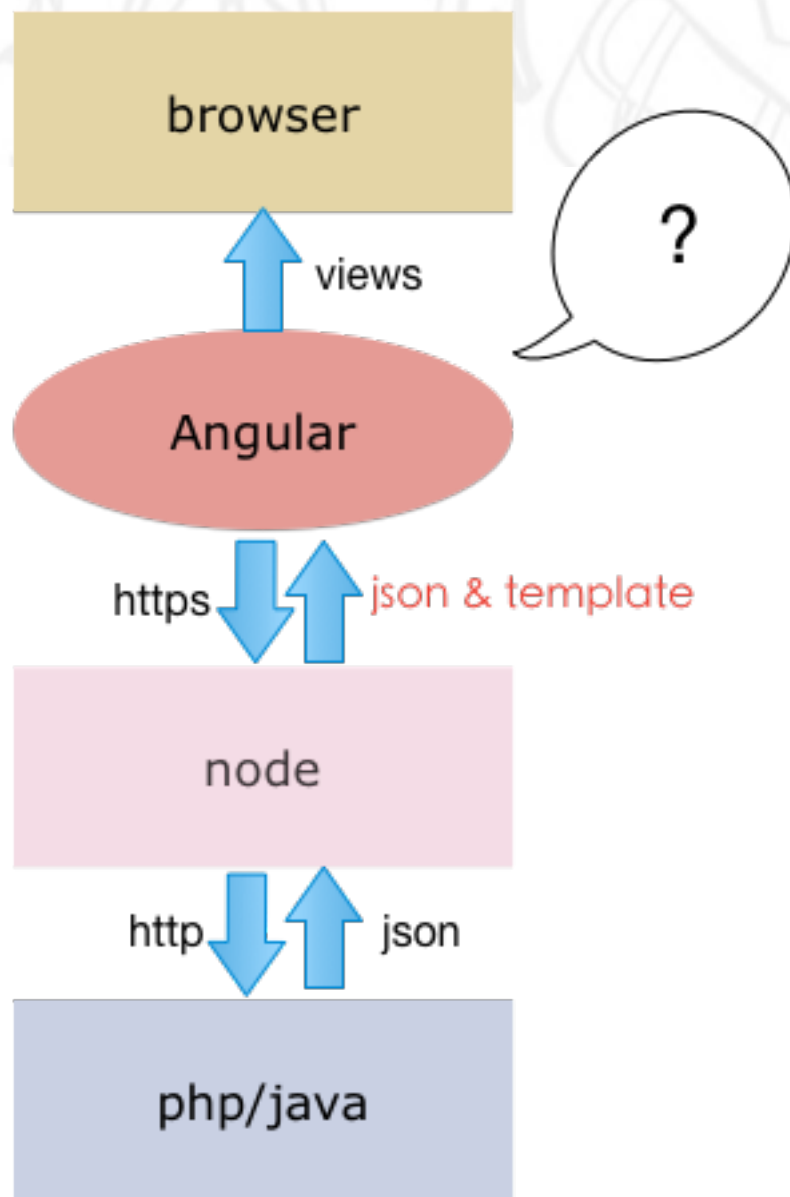
通过eagle我们了解到什么？

- 1、提交代码尽可能写好注释
- 2、尽量code review
- 3、做好容错
- 4、尽可以少次数请求后台接口，该合并的要合并

我们的收获？

- 1、哪个熊孩子在刷接口，一目了然
- 2、双十一，全民疯抢的时候，可以专心的做自己的事情
- 3、真正能发现问题
 - 看到有很多邮件过来，作为团队的一员，很失落。但是做为eagle的开发者的，真的很开心，原谅我如此这般。。。
- 4、让新来的同学练练手。
 - 至少我就是在eagle中找到了node的乐趣。同样可以用在php、java、go等

存在的问题







Thanks !

for sleeping and listening