

# Yii2 集成echarts做可视化展示

Created by 杨超, last modified on 2017 Feb 10

- 数据收集
- 数据展示
  - 数据查询
  - 数据展示
    - 演示效果图
  - 数据解析

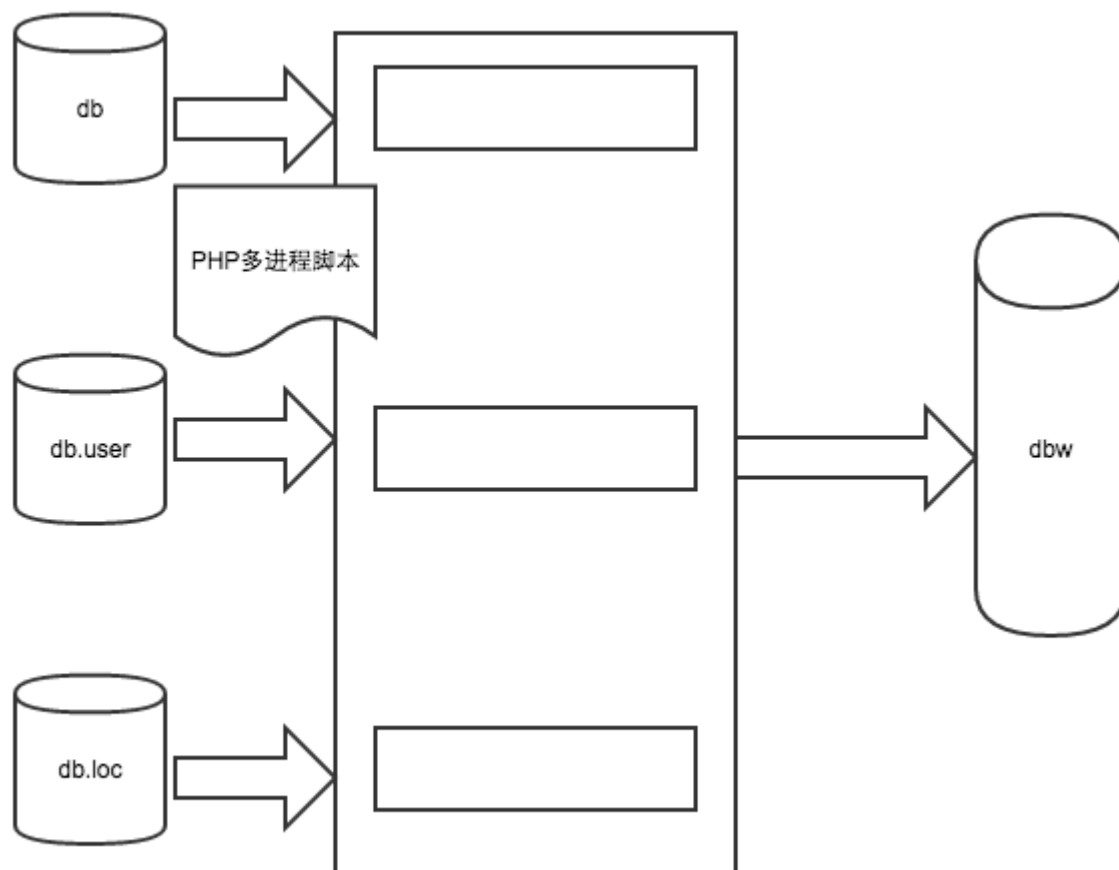
## 数据收集

这部分，**第一期只做数据库的同步**

具体做法如下：每日夜晚24点之后，启动php同步脚本，将数据库中的所有table都同步到 数据仓库的db（姑且叫dbw）中

php同步脚本是一个多进程的脚本，我们通过一个conf配置文件，里面写明现有db的名字以及连接串，然后脚本根据conf中的db数量N 启动N个不同的进程，

去读取每个数据库里的 information\_schema.tables 表，然后依次 根据modtime更新每天的数据 到dbw中，dbw中表的命名遵循原则 库名.表名，例如 db\_order, loc\_city



## 数据展示

### 数据查询

使用纯sql进行查询，建立一张表 sql 存储查询的sql语句，

目前想法是 表里 包含 sql语句，建表时间和建表人，以及一些可选的配置（这个字段是个json串，后期定制化的时候会有用处，一起可以先粗暴使用默认配置，未来逐步的优化

### 数据展示

可选的图形化插件还是挺多的，从 highcharts/charts.js/echarts，有很多，而且都有对应的yii的封装，目前个人倾向于选择echarts，之前在mt用来很久的高charts，对于动态图和map的封装不是太好，而echarts是百度新开发的，3.0之后融合了动态展示和china地图的原生支持，图形效果也不错，考虑先采用echarts

echarts 官网: <http://echarts.baidu.com/index.html>

<http://echarts.baidu.com/echarts2/doc/slide/whyEcharts.html#/>

yii2 echarts封装: <https://github.com/daixianceng/yii2-echarts>

个人编写的demo示范:

#### View-demo

[Collapse source](#)

```
View代码:
<?php
use yii\helpers\Html;
use yii\web\JsExpression;
use daixianceng\echarts\ECharts;
$this->title = '数据展示平台';
?>
<div class="community-view">
    <h1><?= Html::encode($this->title) ?></h1>
    <?= ECharts::widget([
        'responsive' => true,
        'options' => [
            'style' => 'height: 400px;'
        ],
        'pluginEvents' => [
            'click' => [
                new JsExpression('function (params) {console.log(params)}'),
                new JsExpression('function (params) {console.log("ok")}')
            ],
            'legendselectchanged' => new JsExpression('function (params) {console.log(params.selected)}')
        ],
        'pluginOptions' => [
            'option' => [
                'title' => [
                    'text' => $title
                ],
            ],
        ],
    ])
```

```
'tooltip' => [
    'trigger' => 'axis'
],
'legend' => [
    'data' => $entitylist
],
'grid' => [
    'left' => '3%',
    'right' => '4%',
    'bottom' => '3%',
    'containLabel' => true
],
'toolbox' => [
    'feature' => [
        'dataView' => ['show' => true, 'readOnly' => false],
        'magicType' => ['show' => true, 'type' => ['line', 'bar', 'stack', 'tiled']],
        'restore' => ['show' => true],
        'saveAsImage' => []
    ]
],
'xAxis' => [
    'name' => $xField,
    'type' => 'category',
    'boundaryGap' => false,
    'data' => $xRangeValues
],
'yAxis' => [
    'type' => 'value'
],
'series' => $seriesData,
]
]; ?>
</div>
```

## Controller-demo

[Collapse source](#)

```
Controller:
<?php
    public function actionIndex2()
    {
        $title = '每日楼栋下单数';
        $xField = 'datekey';
        $xRangeValues = [
            20160401,
            20160402,
            20160403,
            20160404,
            20160405,
            20160406,
            20160407,
            20160408,
            20160409,
            20160410,
            20160411,
            20160412,
            20160413,
            20160414,
            20160415,
            20160416,
            20160417,
            20160418,
            20160419,
            20160420,
            20160421,
            20160422,
            20160423,
            20160424,
            20160425,
            20160426,
```

```
20160427,  
20160428,  
20160429,  
20160430,  
];  
$yField = 'total';  
$entitylist = [5944, 5948, 5949, 5956, 5950];  
$metaData = ['name'=>'', 'smooth'=>true, 'type'=>'line', 'data'=>[]];  
$datas =  
    Orderstat::find()  
    ->select(['datekey', 'buildingid', 'count(*) `total`'])  
    ->andWhere(['datekey'=>$xRangeValues])  
    ->groupBy(['datekey', 'buildingid'])  
    ->asArray()->all();  
$seriesData = [];  
foreach ($entitylist as $entity) {  
    $seriesData[$entity] = $metaData;  
    $seriesData[$entity]['name'] = $entity;  
    foreach ($xRangeValues as $xvalue) {  
        $seriesData[$entity]['data'][$xvalue] = 0;  
    }  
}  
foreach ($datas as $rows) {  
    $datekey = $rows['datekey'];  
    $buildingid = $rows['buildingid'];  
    $total = $rows['total'];  
    if (!isset($seriesData[$buildingid][$datekey])) {  
        $seriesData[$buildingid]['data'][$datekey] = $total;  
    } else {  
        $seriesData[$buildingid]['data'][$datekey] += $total;  
    }  
}  
foreach ($seriesData as $key=>$arr) {  
    ksort($arr['data'], SORT_NUMERIC);  
    $arr['data'] = array_values($arr['data']);  
    $seriesData[$key] = $arr;
```

```
}  
$seriesData = array_values($seriesData);  
return $this->render('index2', [  
    'title' => $title,  
    'xField' => $xField,  
    'yField' => $yField,  
    // 只能展示string  
    'entitylist' => array_map('strval', $entitylist),  
    'datas' => $datas,  
    'xRangeValues' => $xRangeValues,  
    'seriesData' => $seriesData,  
]);  
}
```

### 演示效果图

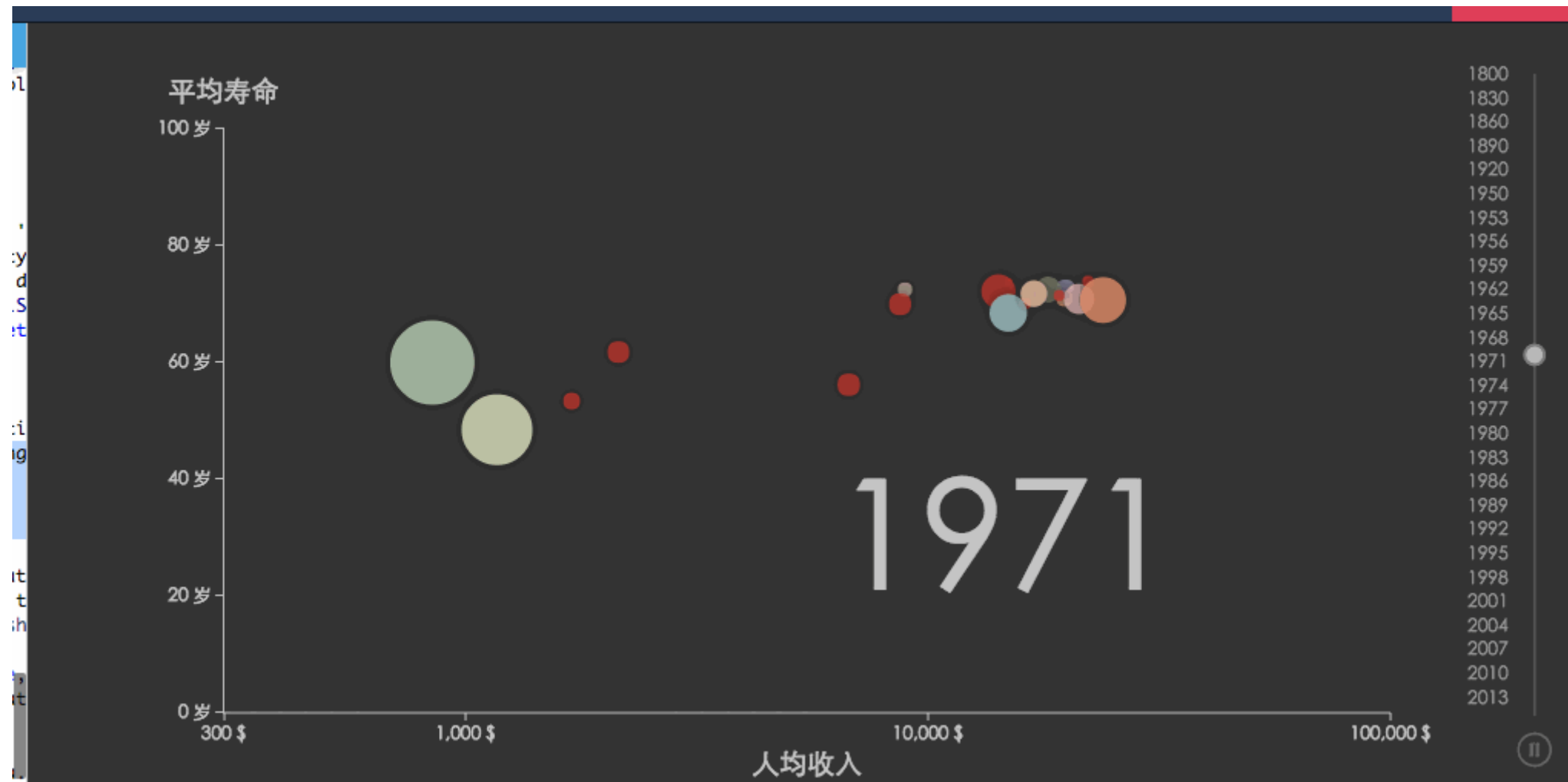
测试link: <http://myc.dev.honglingjinclub.com/loc/index2>

原始demo: <http://myc.dev.honglingjinclub.com/loc/index>



====ECharts 的动态效果图，未来可以很好的支持扩展





## 数据解析

刚才分别阐述了数据查询和展示两个独立的模块的构建想法，但整个的流程要 运转 就必须 将查询的sql 结果 从 查询模块 解析出来 然后再生成图表

这个地方是两者的桥梁，也是最麻烦的一部分

最常见的sql语句

例如我查询一个月的orderstat数据

```
$sql = 'select * from orderstats2 where datekey between 20160401 and 20160430'
```

然后在解析模块中,

1) 要能解析出来 \$table 是 orderstats2

2) 然后直接执行\$sql (当然这里有安全隐患, 即sql注入问题, 可以以后再考虑)

```
$sql = 'select * from orderstat2 limit 3';  
$command = Yii::$app->get('db')->createCommand($sql);  
$rets = $command->queryAll();
```

这样得出的结果是一个数组, 里面含有各个数据库的字段

3) 要知道 x轴和y轴的显示列, 目前的默认做法 可以先以第一列为x轴, 其他为y轴

[Like](#) Be the first to like this

No labels

地址: 北京市朝阳区建国路86号佳兆业广场北塔6层梦想加空间601室

以太资本由艾普拉斯投资顾问(北京)有限公司运营, 提供早期互联网项目的投融资对接服务

©2014-2017 以太资本 京ICP备14028208号