

ASCDC EChart.js TS

整體架構

網頁畫面

Code function

indxe.html

Line 80~89 : class keyword_search

Line 97~99 : JS script

API.js

Main_setting.js

Line 2~17

Line 21~138 echart.js parameter

Line 141~221 JQuery_slider_setting

Line 224~227 sidebar_level_render()

new_data_format.js

Line 2~11 variable

Line 13~128 function data_format(data)

Line 131~133 function calculate_color()

new_Event copy.js

Line 18~23

Line 29~238

Line 245~296 legendselectchanged event

Line 304~374 function max_Level(ui_value)

Line 377~440 JQuery slider function

Line 443~449 function event_setOption_function()

Line 451~472 click event

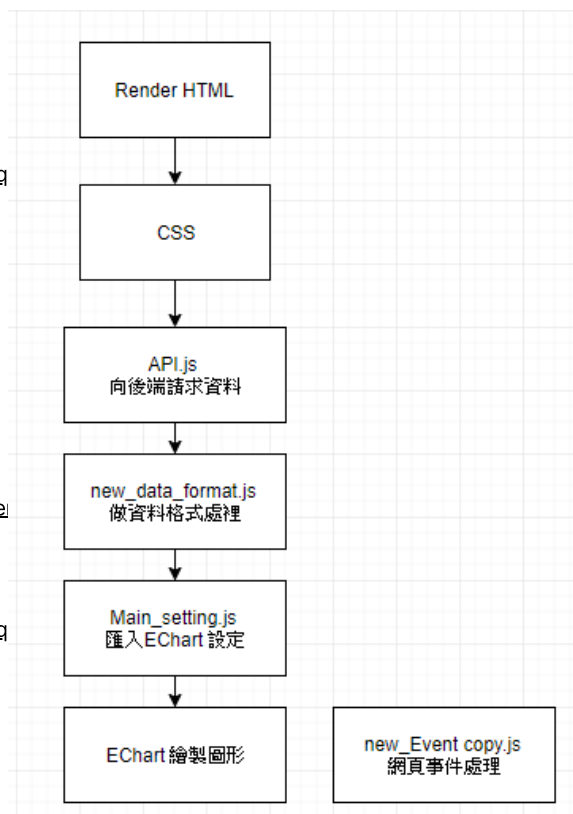
注意事項

Force layout - Mirror edgeLabel problem

目前解決方案

整體架構

- indxe.html
 - 左側sidebar HTML div
- CSS
 - CDN :
<https://cdnjs.cloudflare.com/ajax/libs/jqueryui/1.12.1/jqueryui.css>
 - JQuery-UI CSS 文件
 - index.css
- JavaScript
 - echarts 4.2.0-rc.2.js
 - 主要 function library
 - CDN :
<https://cdnjs.cloudflare.com/ajax/libs/jquery/3.3.1/jquery.js>
 - JQuery library
 - CDN :
<https://cdnjs.cloudflare.com/ajax/libs/jqueryui/1.12.1/jqueryui.js>
 - JQuery-UI library
 - new_data_format.js
 - 從後端接收資料後, 做資料格式處理, 使其格式符合 EChart.js 要求之格式

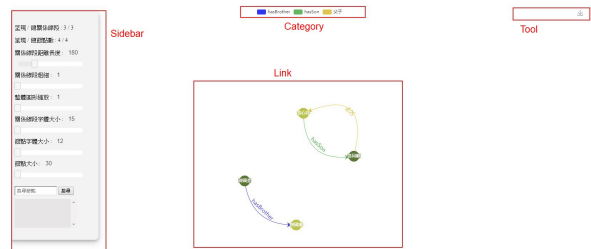


- API.js
 - 向後端請求資料, 並傳送至前端
- Main_setting.js
 - 主要調整 EChart.js 在處理資料視覺化時的設定檔
- new_Event copy.js
 - 負責案件事件的 JS 執行
- Source - 本機端測試資料

流程圖

網頁畫面

- Left Sidebar - 調整主畫面 Link 的線段及節點大小
- Link - 由Ehcart.js 繪製
- Category - 每個 Link type 都繪製個 button, 按下後會影響主畫面的線段顯示
- Tool - 右方save image symbol按下後, 可擷取主畫面圖像



Code function

indx.html

Line 80~89 : class keyword_search

主要用於 keyword search feature,

下方 keyword_item class 會在輸入顯示後新增在 keyword class 裡

```
<div class="keyword_search">
  <input type="text" placeholder="搜尋節點" id="keyword_search_field">
  <button id="keyword_search_button" type="submit">搜尋</button>
  <div class="keyword">
    <!-- <div class="keyword_item">
      <p class="keyword_name">公主</p>
      <button class="keyword_delete">delete</button>
    </div> -->
  </div>
</div>
```

Line 97~99 : JS script

請注意, 因畫面上方 category button 本事設計給node name 並非 link type, 所以有調整了EChart.js 使其不跳出 console.warning, 並不會造成程式上運行問題

```
<!-- avoid echarts.js library console.warn, use the download file and common
<!-- <script src="https://cdnjs.cloudflare.com/ajax/libs/echarts/4.2.0-rc.2.js">
<script src="./lib/echarts 4.2.0-rc.2.js"></script>
```

API.js

這支JS 主要會由後端做調整, 無須刻意修改參數

Main_setting.js

Line 2~17

```
$('.sidebar').hide();
const Chart = echarts.init(document.getElementById('main'),
```

2 & 17 : 在尚未處理資料且尚未繪製圖形時, 先將 sidebar 隱藏

3~7 : 初始化echart 所需要版面

10 & 16 : 在尚未處理資料時, 顯示 loading symbol (會因資料量過大而影響持續動畫時間)

12~15 : 向後端請求接收資料並對其處理格式

```
renderer: 'canvas', //could be svg, but may have perform
width: 1600,
height: 900 //有置中問題
});
var jdata; // ! api.js variable

Chart.showLoading();
// jQuery ajax data from back-end
const api_url = API_generator(2207, 56, 54, 1000);
GetJSON(api_url);
const data = data_format(jdata);
console.log(data);
Chart.hideLoading();
$('.sidebar').show();
```

Line 21~138 echart.js parameter

版面問題又方程式碼不全部放上, 參數部分請參照

<https://www.notion.so/ASCDC-EChart-js-TS-5a807845ad1941cda6a0c065e64e4f1e#8e2a7d21aed44c69898e65a0a752f4ca>

21 : 宣告 option object 存放 parameter

137 : call Chart.setOption 來 render 主畫面

138 : 此function 會影響到 left sidebar 上方的數值

```
var option = {
}
// ! setup all option
Chart.setOption(option);
sidebar_level_render();
```

Line 141~221 JQuery_slider_setting

此Array 主要會傳給 new_Event copy.js 作為 設定 jQuery slider parameter.

object : 對印到 slider 之名稱

min : slider 數值最小值

max : slider 數值最大值

step : 每條整一格數值變動數

value : 目前顯示之數值

disabled & range : 目前似乎無法起作用

```
var jquery_slider_setting = [{
  object: 'max_level',
  min: 1,
  max: 5,
  step: 1,
  value: 1, //current option setting value
  disabled: true,
  range: 'min'
}, {
```

Line 224~227

sidebar_level_render()

處理統計畫面 node and link 數量

```
function sidebar_level_render() {
  $('.current_relation').html(`呈現 / 總關係線段 : ${option.series[0].links.length}`);
  $('.current_node').html(`呈現 / 總節點數 : ${option.series[0].nodes.length}`);
};
```

new_data_format.js

Line 2~11 variable

宣告變數, 會存進 buf 裡後輸出給

main_setting.js 使用

```
var all_nodes = [];
var color = {};
var buf = { //暫存用
  nodes: [],
  all_nodes: [],
  links: [],
  category: [],
  all_category: [],
};
var random_color;
```

Line 13~128 function data_format(data)

版面問題又方程式碼不全部放上

主要將後端傳回之資料全部跑過, 並將每個node 和 link 輸出

15 : 宣告 new set, 之後會用於刪除多重搜尋結果

17 : 開始從每個K1 開始跑

19~24 : 由parameter gp 來分組顏色, 若color 並沒有gp 相對應的顏色, 則產生一個

27~35 : 相K1 push進buf.nodes, 所有點都必須有產生

38 : 從 kg2 開始跑

40~47 : 由parameter type來分組顏色, 若color 並沒有type 相對應的顏色, 則產生一個

49~54 : 由parameter gp 來分組顏色, 若color 並沒有gp 相對應的顏色, 則產生一個

56~65 : 相K2 push進buf.nodes, 所有點都必須有產生

67~85 : 將 target and source 寫進 category, 以產生link

102 : link 和 category 所需資料相同

106~108 : 去除相同值

110~113 : 相node.name 寫入all_nodes 以利後續keyword search 判斷是否正確

115~118 : category 按照名稱排序

```
const set = new Set(); //為了之後去除重複

data.forEach(data_element => {
  //判斷有沒有顏色 沒有的話隨機產生
  if (color.hasOwnProperty(data_element.gp)) {
    node_color = color[data_element.gp]
  } else {
    node_color = calculate_color();
    color[data_element.gp] = node_color;
  }

  // push k1 into nodes array
  buf.nodes.push({
    name: data_element.k1,
    gp: data_element.gp,
    itemStyle: {
      normal: {
        color: node_color
      }
    }
  });
});
```

Line 131~133 function calculate_color()

產生顏色

```
function calculate_color() {
  return '#' + (Math.random() * 0xFFFFFF << 0).toString(16);//<<:左移賦值
}
```

new_Event copy.js

Line 18~23

18~21 : 針對keyword search 做事件綁定

23 : 若現再並沒有搜尋關鍵字, 則max level 隱藏起來

```
$('#keyword_search_field').keyup((e) => {
  keyword_search(e)
});
$('#keyword_search_button').on('click', keyword_search);

keyword.length == 0 ? $(".max_level").hide() : $(".max_level
```

Line 29~238

31 : 取得 keyword_search_field 文字

34~36 : 觸發事件去判斷此輸入的關鍵字是否有在資料內

42~45 : 顯示相同字不可重複輸入

60 function keyword_item_append : 產生相對應的 div 並append 到 HTML上

69 function data_filter() :

95 : 需先判斷此時max level slider 數值為多少, 先以一為例

99~101 : 若category.target or source 和 keyword 相同, 則篩選到category, 會取用其type

103~111 : 由 category buton 是否點選做判斷, 若category.show 為 true, 則輸入至links

113~114 : 刪除相同值, 以利後續nodes篩選

```
union_collect = [];
$('#max_level').val() === '1' ? data_filter_max_level_1() : data_fil

function data_filter_max_level_1() {
  let union_collect_category_show = [];
  option.series[0].categories = data.category.filter(category => {
    return keyword.includes(category.target) || keyword.includes
  });

  option.series[0].links = option.series[0].categories.filter(cate
  union_collect.push(category.target, category.source);
  // return category.show == true;

  if(category.show == true){
    union_collect_category_show.push(category.target, category
```

120~124 : 若node 和 keyword值相同, 則顏色改成紅色,

128 function data_filter_max_level_2()

和上述相同, 只不過需要以階層數向外擴展, 由每個節點所連結出的點作為出發點, 從以格式化的資料中找尋是否有相同的 Target or source , 並做篩選

139 function keyword_item_delete()

195 : 會發生重複綁定delete 的事件, 必須先解綁後再重新綁定事件才可

210 function keyword_search_reset()

和上述解釋一樣

```
        return category;
    }
});

union_collect = Array.from(new Set(union_collect));
union_collect_category_show = Array.from(new Set(union_collect_category_show));

console.log(union_collect);
console.log(union_collect_category_show);
// console.log(color)

option.series[0].nodes = data.nodes.filter(node => {
    keyword.includes(node.name) ? node.itemStyle.normal.color =
        return union_collect.includes(node.name) && union_collect_category_show.includes(node.name) ? node.itemStyle.normal.color = 'red' : node.itemStyle.normal.color;
});
}
```

Line 245~296 legendselectchanged event

主要判斷為上方 category button 的點選, 依據點選結果會造成主畫面的 link 顯示與否, 程式邏輯和以上基本都一樣

Line 304~374 function max_Level(ui_value)

以max level slider 為判斷依據, 程式邏輯和以上基本都一樣

Line 377~440 jQuery slider function

此段是主要先從 main setting .js 提取預先設定好的 jQuery slider 數值並透過 slider function 來產生 sidebar slider 每一次拉動都會造成主畫面的調整

Line 443~449 function event_setOption_function()

此段主要配合上面 jQuery slider function 的設定, 在每次拉動 slider 後Call 這function , 並更新主畫面

Line 451~472 click event

此段是點選主畫面 link type 時, 會跳轉到對相對應的文本網頁位置

注意事項

Force layout - Mirror edgeLabel problem

發文請點選此 github issues , 我在先前有向開發團隊問過此問題

<https://github.com/apache/incubator-echarts/issues/10643>

目前解決方案

修改echarts 4.2.0-rc.2.js @2271~2305, 新增 function label_rotate(out, a, rad, sub_rotation) 來處理此問題, 此段會以 rad作為判斷依據, 並調整之中 type 文字的旋轉方向.