拓扑图组件预研

### 应用背景

在产品应用中，不同层级、相同层级之间的隶属和相互关系（拓扑）需要在前端界面进行直观的展示，并能够对其做简单的操作（编辑、拖拽，连接等）。

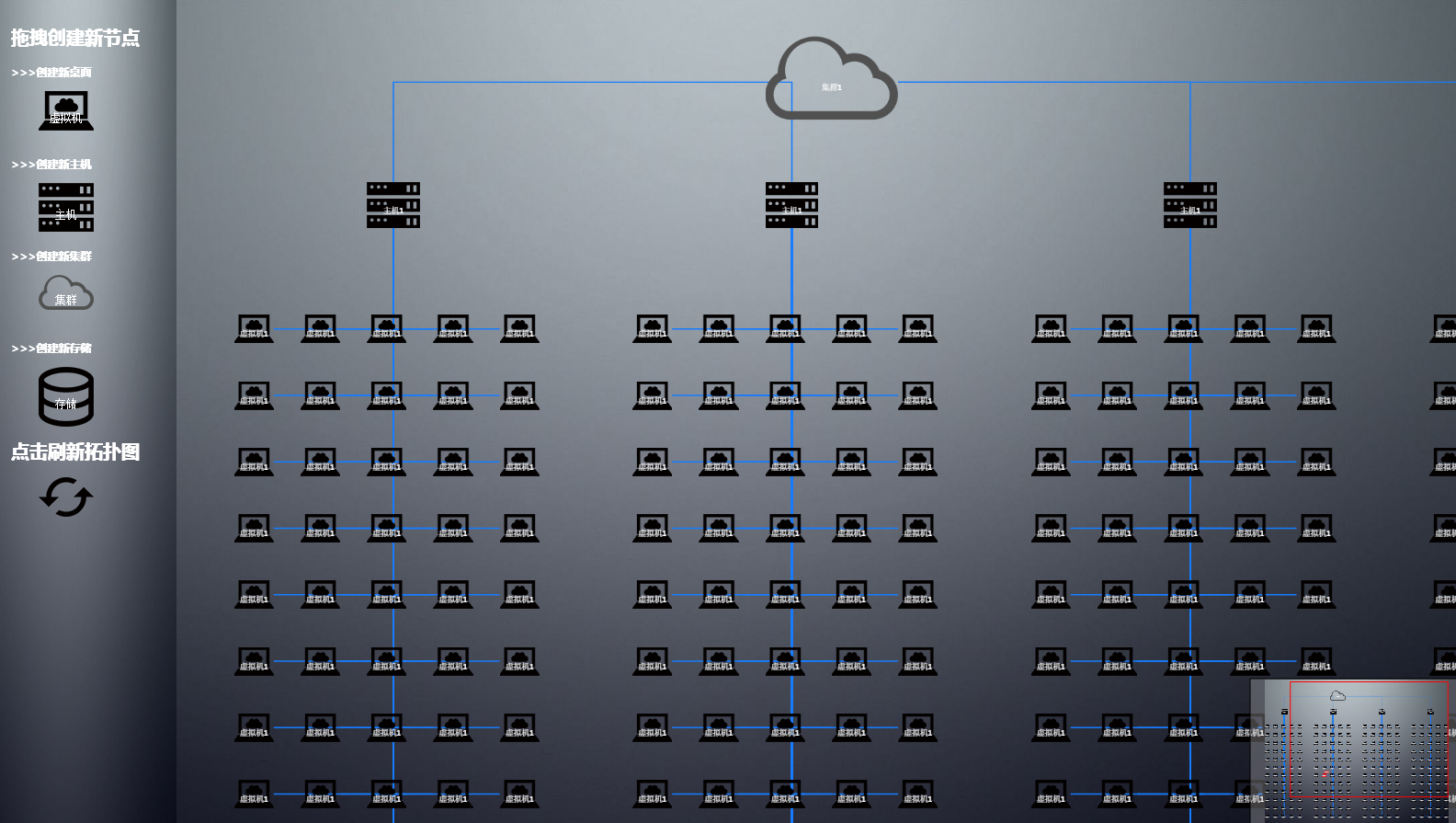
### 技术选择

HTML5支持SVG和CANVAS进行绘图，结合IVYCloud之前的折线图、环形进度条等都是基于canvas做的，这里选择canvas作为技术实现方案。

#### 框架

基于Html5 Canvas的拓扑函数库比较少，比较常见的有Qunee/Hightopo/Jtopo，其中Qunee和Hightopo都是商用收费（没有搜索到源文件，故暂不考虑）。Jtopo是国内自主设计的完全基于HTML5 Canvas的使用javascript纯脚本语言的关系、拓扑图形化开发工具包，支持多平台（HTML5）、跨浏览器,文档资料比较齐全，不依赖出jquery之外的其他库，执行仅需一个canvas，不污染界面、DOM结构和代码命名空间，考虑到上述，此处选取Jtopo作为预研框架选型（Jtopo也只有jtopo.min.js文件，没有搜索到源文件，此处通过反压缩得到可读性较强的源码）。

##### 整体效果



##### 应用方法

###### 引入库：



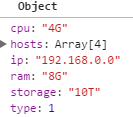
###### 创建jtopo对象



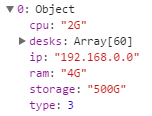
###### 传入更新数据

传入的数据以对象数组的形式，在总节点（集群）中包含了子节点（主机）数组，在主机节点中包含了桌面数组；其数据如下图：

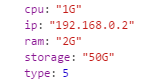
集群对象：



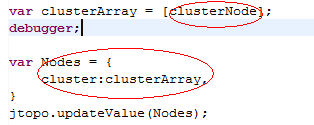
主机对象：



虚拟机对象：



最终，各个层级的对象数组组合成json数据，传入组件更新操作：



###### 在组件中菜单栏点击更新按钮触发事件的处理：



##### Jtopo中的基本元素介绍

Canvas

Canvas是绘图的基础，即画布

Stage

Stage是基于canvas的“舞台”对象,jtopo中基于stage进行scene的显示

Scene

基于stage的“场景”对象

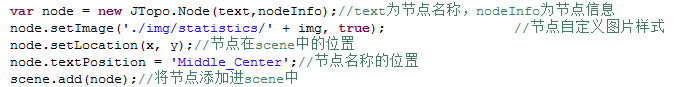
Node

Jpoto中的节点对象

Link

Jtopo中用来连接不同Node

Jtopo创建节点：



Jtopo连接节点

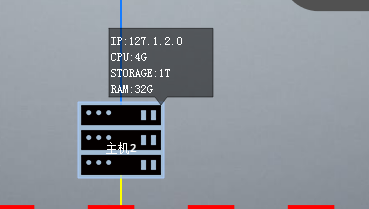


##### 节点鼠标事件

###### 节点信息展示：

通过捕捉到鼠标的悬停事件，进而判断当前位置是否节点并获取当前节点的信息加以在canvas中展示：

效果图：

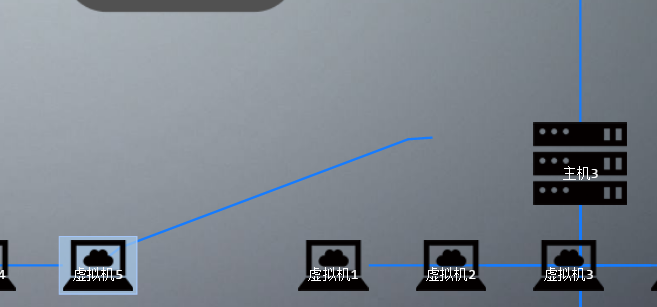


###### 节点拖拽（子节点相应跟随拖拽）:

通过捕捉鼠标拖动事件的头尾位置，对该节点及其下属节点做位置偏移设置（在节点连接的时候对于相关的两个节点分别记录其子节点和父节点集）。

###### 节点连接：

判断鼠标点击事件，如果点击事件的target为node，且其后的点击事件的target也为node，则在两个node之间新建link（中间可以根据节点信息做判断，是否连接以及连接类型）。



###### 节点右键功能



结合div、label和canvas元素，在节点右键点击的时候将隐藏的选择菜单显示出来，并设置其位置在节点附近。通过判断不同label的点击事件，分别进行处理。

###### 快速删除节点功能

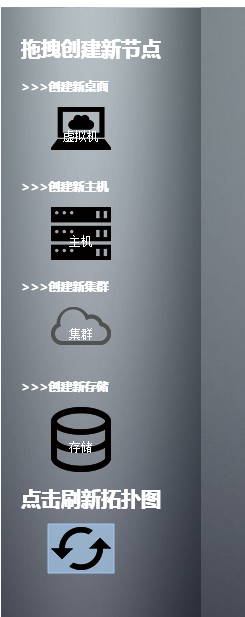
在右键菜单栏的点击事件中，若是点击删除节点，则删除所对应的节点

###### 节点编辑



在右键菜单栏中点击编辑，依旧是结合div、label、input和canvas元素，通过绘制表单元素以及定位到节点周围的方式，实现对节点的信息编辑。

##### 边菜单视图栏



###### 菜单栏和拓扑图移动和缩放问题的处理：

在右边拓扑图的绘制和展示、查看过程中，需要实现放大缩小以及移动效果，但是菜单栏中的元素应该是禁止移动和缩放的。如果单张画布的话，实现同一画布不同区域的缩放控制和移动比较复杂。采用两张画布的方式，针对菜单栏的画布禁止了移动和缩放功能。

###### 拖拽新建节点

通过拖拽的方式，实现新增所需要的虚拟机、主机、集群、存储等节点。此处的实现方式是在拖拽菜单栏中的节点的同时在原处新建节点，实现无限制的新建节点。当菜单栏画布中所拖拽的新节点超过画布范围，则触发相应时间，隐藏当前新节点，在拓扑图画布中新建相应类型的新节点，并通过鼠标的move事件实现该新节点跟随鼠标移动变化位置，实现了从菜单栏画布到拓扑图画布的流畅过度。并按照实际需求更换拓扑图画布中节点的背景图片以适应实际图谱图。

###### 刷新拓扑图

单击按钮触发更新拓扑图事件，根据对外接口updateValue提供的新的节点信息进行对拓扑图进行重绘（具体操作流程为：①在点击按钮时由组件抛出事件，由调用组件的程序捕捉该事件，触发相应更新。②remove现有的所有节点③按照节点信息重绘节点）；