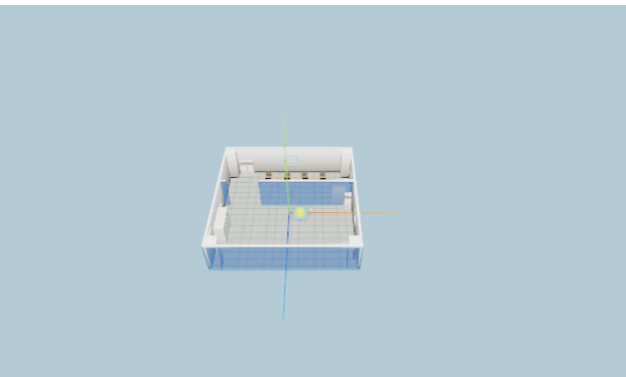


plugin-heat-map

npm@latest v2.9.10

热力图。

样例



重置数据
第二个热力图
通过id获取
通过名称获取
通过id移除
Close Controls

objects 31
triangles 28736
vertices 86206
frametime 0.50 ms



安装

```
npm install @soonspacejs/plugin-heat-map -S
# or
yarn add @soonspacejs/plugin-heat-map -S
```

sh

使用方法

☰ SoonSpace.js 2.x

```
import { HeatMapPlugin } from '@soonpacejs/plugin-heat-map';

const ssp = new SoonSpace({
  el: '#view',
  options: {},
  events: {},
});

const heatMap = ssp.registerPlugin(HeatMapPlugin, 'heatMap');
console.log(heatMap);
```

方法

create

创建热力图

定义

```
interface SceneDataPoint {
  x: number;
  z: number;
  value: number;
  radius: number;
}

interface CreateParam {
  // base
  id: string;
  name?: PluginObject['name'];
  data: SceneDataPoint[];
  // position
  yAxisHeight: number;
  minPosition: PlaneIVector2;
  maxPosition: PlaneIVector2;
  // value
  min?: number;
  max?: number;
  radius?: number;
}
```

ts

☰ SoonSpace.js 2.x

```
function create(param, createParam): PluginObject,
```

用法:


```
heatMap.create({
  id: 'hm1',
  name: 'hm_name',
  yAxisHeight: 100,
  max: 100,
  min: 1,
  minPosition: {
    x: 0,
    z: 0
  },
  maxPosition: {
    x: 200,
    z: 200
  },
  data: [
    {
      x: 100,
      z: 100,
      radius: 100,
      value: 80
    },
    {
      x: 200,
      z: 50,
      radius: 50,
      value: 80
    },
    {
      x: 100,
      z: 50,
      radius: 50,
      value: 80
    }
  ]
})
```

js

☰ SoonSpace.js 2.x

参数：

param

- 描述: 创建热力图参数
- 必填: 
- 类型: `CreateParam`

CreateParam

属性	描述	类型	必填	默认值
id	热力图对象 ID	string		
name	热力图对象名称	string		
data	热力图数据	<code>DataPoint</code>		
max	数据中单点值大于等于该值时，以最深热力颜色展示	number		100
min	数据中单点值小于等于该值时，以最浅热力颜色展示	number		1
radius	热力点半径	number		100
yAxisHeight	空间高度	number		
minPosition	热力图绘制区域最小点	<code>PlaneIVector2</code>		
maxPosition	热力图绘制区域最大点	<code>PlaneIVector2</code>		

DataPoint

属性	描述	类型	必填	默认值
x	点位在空间平面内的水平位置	number		
z	点位在空间平面内的垂直位置	number		
value	热力值	number		

☰ SoonSpace.js 2.x

radius	热力点半径	number	✗	100
--------	-------	--------	---	-----

注意

`x`、`z` 的位置需要在 `minPosition` 和 `maxPosition` 之间，否则不会渲染

`radius` 的显示范围会随着画布区域放大与缩小

setData

设置（重置）数据

定义：

```
function setData(id: CreateParam['id'], data: SceneDataPoint[]): PluginObject
```

用法：

```
heatMap.setData(  
  'hm1',  
  [  
    {  
      x: 100,  
      y: 100,  
      radius: 100,  
      value: Math.floor(Math.random() * 100)  
    },  
    {  
      x: 200,  
      y: 50,  
      radius: 50,  
      value: Math.floor(Math.random() * 100)  
    },  
  ],  
)
```

☰ SoonSpace.js 2.x

```
radius: 50,  
value: Math.floor(Math.random() * 100)  
},  
]  
)
```

参数:

id

- 描述: 已创建热力图的 id
- 必填: ✓
- 类型: `CreateParam['id']`

data

- 描述: 新点位数据
- 必填: ✓
- 类型: `SceneDataPoint[]`

createPolygon

创建一个具有多边形边界的热力图。

通过 `points` 选项来设置多边形的顶点的世界坐标。

它会根据你传入的顶点，来创建多边形边界，该多边形边界会以前3个顶点所确定的平面来作为多边形边界的所在平面，对于那些不与该多边形共面的点，会向该平面上投影，然后将最终的投影点作为多边形边界的顶点。

热力图所需的数据点也是世界坐标系下的三维坐标，它会自动将数据点转换成在多边形平面上的投影点并将该投影点作为热力图最终的数据点。

样例

移动多边形热力

反向移动多边形

Close Controls

objects 29
triangles 28734
vertices 86202
frametime 13.80 ms



与 create() 对比

用户在使用热力图时，一般期望的逻辑是：

- 传给热力图的数据都是世界坐标系下的数据，这样用户不必做数据的转换
- 可以自定义热力图的边界，因为经常会在不规则的区域内绘制热力图，比如：地铁的站台层、大厅 或 非矩形的房间等。
- 如果移动了热力图对象，原来的热力点的数据也能正确映射。

在这种场景下，使用 create() 创建热力图会有以下缺点：

- 热力图区域只能是 水平的 且是 矩形的，不可以是任意多边形，用户只能手动旋转来让水平的热力图变得倾斜，但这样的话，用户又必须自己将热力点数据从世界坐标系转到 倾斜状态下的局部坐标系。
- 热力图的数据点是 二维局部坐标，用户需要将世界坐标系的点转为二维的 且是 局部的坐标（如果有对热力图对象被移动、旋转或缩放等变换之后）。
- 如果热力图绘制好后，更改了位置、旋转，则用户需要重新将世界坐标系下的热力点针对新的位置和旋转再次进行转换。

createPolygon() 就是为了解决上述缺点而生，所以 createPolygon() 具备以下特点

- 用户可以传递世界坐标系下的三维顶点列表 points 来作为热力图的绘制区域，所以热力图区域可以是任意多边形 且 可以是斜着的。
- 热力图的数据点是 三维的世界坐标，用户不需要额外的转换。

≡ SoonSpace.js 2.x

定义

```
interface ScenePolygonDataPoint {  
  x: number;  
  y: number;  
  z: number;  
  value: number;  
  radius: number;  
}  
  
export interface CreatePolygonParam {  
  id: string;  
  name?: PluginObject['name'];  
  data: ScenePolygonDataPoint[];  
  points: IVector3[];  
  min?: number;  
  max?: number;  
  radius?: number;  
}  
  
createPolygon ( param: CreatePolygonParam ): PluginObject;
```

用法:

```
heatMap.createPolygon({  
  id: 'hm1',  
  name: 'hm_name',  
  max: 100,  
  min: 1,  
  points:[  
    {  
      x: 0,  
      y: 0,  
      z: 0,  
    },  
    {  
      x: 0,  
      y: 0,  
      z: 100,  
    },  
  ],  
})
```


☰ SoonSpace.js 2.x


```
      y: 100,
      z: 100,
    },
  ],
  data: [
    {
      x: 100,
      y: 100,
      z: 100,
      radius: 100,
      value: 80
    },
    {
      x: 200,
      y: 200,
      z: 50,
      radius: 50,
      value: 80
    },
    {
      x: 100,
      y: 100,
      z: 50,
      radius: 50,
      value: 80
    },
  ],
])
})
```

参数:

param

- 描述: 创建热力图参数
- 必填: 
- 类型: CreatePolygonParam

CreateParam

属性	描述	类型	必填	默认值
id	热力图对象 ID	string		

SoonSpace.js 2.x

name	热力图对象名称	string	✗	
data	热力图数据	ScenePolygonDataPoint[]	✓	
max	数据中单点值大于等于该值时，以最深热力颜色展示	number	✗	100
min	数据中单点值小于等于该值时，以最浅热力颜色展示	number	✗	1
radius	热力点半径	number	✗	100
points	多边形的顶点列表	IVector3[]	✓	

ScenePolygonDataPoint

属性	描述	类型	必填	默认值
x	点在世界坐标系下3维坐标的 x	number	✓	
y	点在世界坐标系下3维坐标的 y	number	✓	
z	点在世界坐标系下3维坐标的 z	number	✓	
value	热力值	number	✓	
radius	热力点半径	number	✗	100

注意

如果坐标点在多边形平面的投影点不在多边形区域内的话，就不会渲染该热力点

`radius` 的显示范围会随着画布区域放大与缩小

setDataPolygon

设置（重置）数据

☰ SoonSpace.js 2.x

定义:

```
setDataPolygon ( id: CreateParam['id'], data: CreatePolygonParam['data'] ): Pts
```

用法:

```
heatMap.setDataPolygon(  
  'hm1',  
  [  
    {  
      x: 100,  
      y: 100,  
      z: 100,  
      radius: 100,  
      value: Math.floor(Math.random() * 100)  
    },  
    {  
      x: 200,  
      y: 200,  
      z: 50,  
      radius: 50,  
      value: Math.floor(Math.random() * 100)  
    },  
    {  
      x: 100,  
      y: 100,  
      z: 50,  
      radius: 50,  
      value: Math.floor(Math.random() * 100)  
    },  
  ],  
)  
js
```

参数:

id

- 描述: 已创建热力图的 id

☰ SoonSpace.js 2.x

data

- 描述: 新点位数据
- 必填: ✓
- 类型: `ScenePolygonDataPoint[]`

getById

通过 `id` 创建热力图

用法:

```
heatMap.getById('hm1');
```

js

参数:

- id: string

getName

通过 `name` 创建热力图

用法:

```
heatMap.getName('hm_name');
```

js

参数:

- name: string

removeById

通过 `id` 删除热力图

☰ SoonSpace.js 2.x

用法：

```
heatMap.removeById('hm1');
```

js

参数：

- id: string