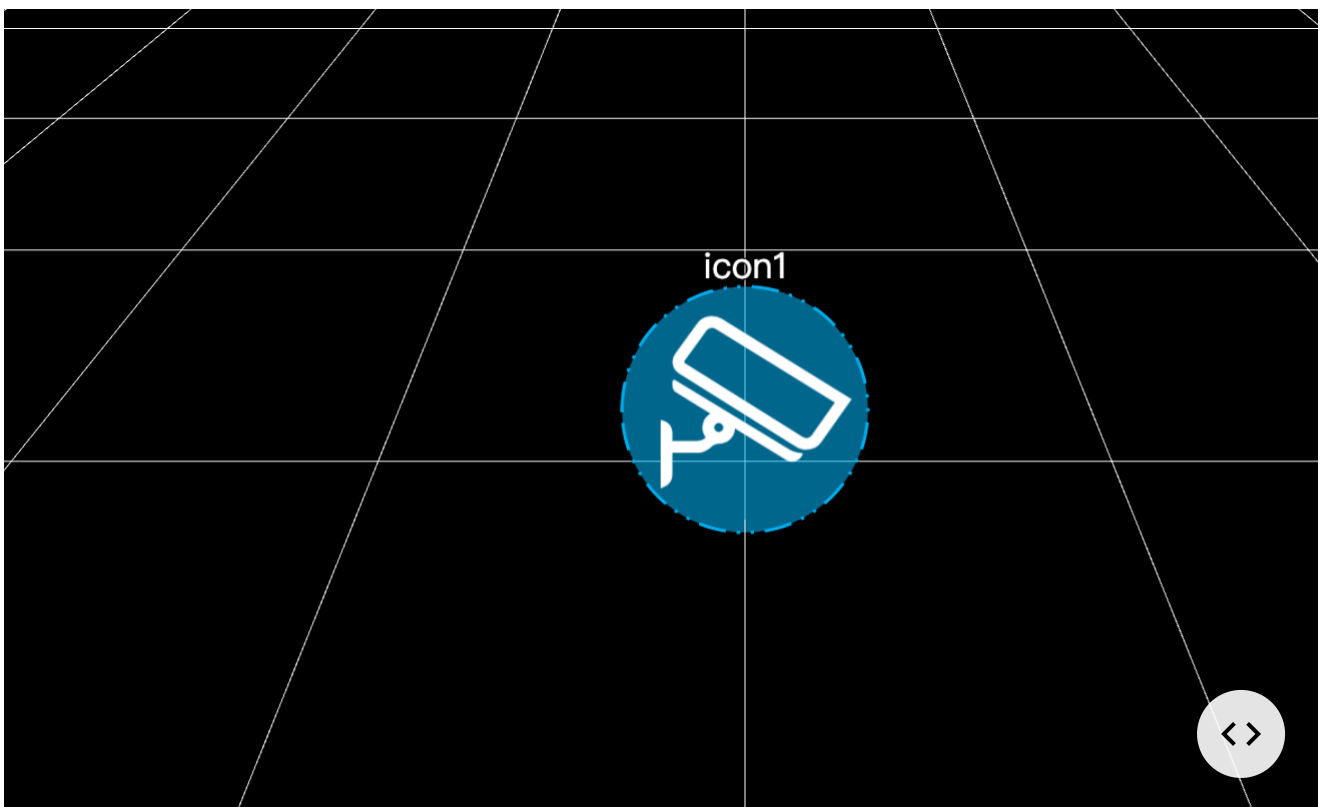


Poi 对象

createPoi

创建 `poi` 对象。

样例：



定义：

```
interface PoiInfo extends BaseObject3DInfo, ObjectEvents<Poi> {  
  url: string;  
  nameScale?: Scale;  
}  
  
function createPoi(poiInfo: PoiInfo): Poi;
```

ts

用法:

```
ssp.createPoi(  
  // poiInfo  
  {  
    id: 'xx',  
    name: 'xx',  
    url: 'http://xx.com/xx.png',  
    level: {  
      max: 1000,  
      min: null,  
    },  
    position: { x: 0, y: 0, z: 0 },  
    rotation: { x: 0, y: 0, z: 0 },  
    scale: { x: 2, y: 2, z: 2 },  
    onClick(e) {  
      /**  
       * 对象的独立事件触发后，默认不传播（类似 DOM 的事件冒泡）到全局事件，  
       * 调用 eventPropagation 方法通知事件继续传播到全局。  
       *  
       * warn:  
       * 在 **非箭头函数** 中参数 e 与 this 的指向都是当前对象对象，  
       * 在 *箭头函数* 参数 e 依然是对象对象，但 this 指向会发生变化。  
       */  
      this.eventPropagation();  
  
      console.log('对象自身的点击事件触发', this);  
    },  
    onDbClick: (e) => {  
      /**  
       * 这里模拟在 **箭头函数** 中  
       */  
      e.eventPropagation();  
  
      console.log('对象自身的双击事件触发', e);  
    },  
    userData: {},  
  }  
);
```

js



参数:

poiInfo

- 描述: 实例 `Poi` 对象所需信息
- 类型: `poiInfo`
- 必填:

C

属性	描述	类型	必填	默认值
<code>id</code>	唯一ID	<code>string</code> <code>number</code>		
<code>name</code>	名称	<code>string</code>		
<code>nameScale</code>	展示名称的缩放比例	<code>Scale</code>		<code>{ x: 16, y: 16, z: 1 }</code>
<code>url</code>	图片资源路径	<code>string</code>		
<code>level</code>	显示层级范围	<code>Level</code>		<code>{ max: null, min: null }</code>
<code>position</code>	位置坐标	<code>Position</code>		<code>{ x: 0, y: 0, z: 0 }</code>
<code>rotation</code>	旋转弧度	<code>Rotation</code>		<code>{ x: 0, y: 0, z: 0 }</code>
<code>scale</code>	缩放比例	<code>Scale</code>		<code>{ x: 1, y: 1, z: 1 }</code>
<code>userData</code>	用户数据	<code>any</code>		<code>{}</code>
<code>onHover</code>	鼠标悬浮事件	<code>Function</code>		<code>null</code>
<code>onUnHover</code>	鼠标悬浮后离开事件	<code>Function</code>		<code>null</code>
<code>onClick</code>	左键单击事件	<code>Function</code>		<code>null</code>
<code>onDbClick</code>	左键双击事件	<code>Function</code>		<code>null</code>

☰ SoonSpace.js 2.x



onRightClick	右键单击事件	Function	✗	null
onLoad	创建完成事件	Function	✗	null

getPoiById

通过 `id` 查找

定义:

```
function getPoiById(id: PoiInfo['id']): Poi | null;
```

ts

用法:

```
const poi = ssp.getPoiById('xxx');
```

js

getPoiByName

通过 `name` 查找

定义:

```
function getPoiByName(name: string): Poi[];
```

ts

用法:

```
const poiList = ssp.getPoiByName('xxx');
```

js

getAllPoi

获取所有 `Poi` 对象

定义:

```
function getAllPoi(): Poi[];
```

ts

用法:

```
const allPoiList = ssp.getAllPoi();
```

js

getPoiByUserDataProperty

通过 `userData` 属性查找

定义:

```
function getPoiByUserDataProperty(  
  propNameOrFindFunc: string | UserDataPropertyFindFunc,  
  value?: any  
): Poi[];
```

ts

用法:

```
const poiList = ssp.getPoiByUserDataProperty('propKey', 'propVal')  
// or  
const poiList = ssp.getPoiByUserDataProperty(item => item['itemPropKey'] ===
```

js

参数:

propNameOrFindFunc

- 描述: `userData` 内属性名 或 `find` 函数
- 类型: `string` | `function`
- 必填: ✓

propValue

- 描述: `userData` 内属性值。
- 类型: `any`
- 必填: ✗

find 函数使用场景

```
poi.userData = {  
  people: {  
    name: 'xiaoming',  
    age: 18,  
  },  
};  
  
const poiList = ssp.getPoiByUserDataProperty(  
  (userData) => userData?.people?.name === 'xiaoming'  
);
```

js

removePoiById

通过 `id` 移除

定义:

```
function removePoiById(id: PoiInfo['id']): boolean;
```

ts

用法:

```
ssp.removePoiById('xxx');
```

js

createPoiToGroup

创建 `poi` 到一个组内。

定义:

```
function createPoiToGroup(groupInfo: GroupInfo, poiInfoList: PoiInfo[]): GroupInfo
```

ts

用法:

```
ssp.createPoiToGroup(  
  // groupInfo  
  {  
    id: 'firstPoiGroup',  
    name: 'name_firstPoiGroup',  
    // ...  
  },  
  // poiInfoList  
  [poiInfo1, poiInfo2, poiInfo3]  
);
```

js

参数

groupInfo

- 描述: 实例组对象所需信息
- 类型: `GroupInfo`

poiInfoList

- 描述: `poiInfo` 集合
- 类型: `poiinfo[]`
- 必填: ✓

createGroupForPoi

为 `poi` 提前创建一个空组。

使用场景

与 `createPoiToGroup` 不同，有些时候可能你还没有具体的 `poiInfo` 数据，但你想提前创建一个批量管理的空组，当有数据时再使用 `addPoiForGroup` 插入。

定义：

```
function createGroupForPoi(groupInfo: GroupInfo): Group;
```

ts

用法：

```
ssp.createGroupForPoi({  
  id: 'firstPoiGroup',  
  name: 'name_firstPoiGroup',  
  // ...  
});
```

js

参数

groupInfo

- 描述: 实例组对象所需信息
- 类型: `GroupInfo`

addPoiForGroup

向一个已经存在的组内添加 `poi` 对象。

定义:

```
function addPoiForGroup(  
  groupId: GroupInfo['id'],  
  poiInfoList: PoiInfo[]  
): Group | null;
```

ts

用法:

```
ssp.addPoiForGroup(  
  // groupId  
  'firstPoiGroup',  
  // poiInfoList  
  [poiInfo4, poiInfo5],  
  // onProgress  
  (progress) => console.log('进度信息: ', progress)  
);
```

js

参数

groupId

- 描述: 组 `id`
- 类型: `groupId['id']`
- 必填: ✓

poiInfoList

- 描述: `poiInfo` 集合
- 类型: `poiinfo[]`

getPoiGroupById

通过 `id` 查找 `poi` 组

定义:

```
function getPoiGroupById(id: GroupInfo['id']): Group | null;
```

ts

用法:

```
const group = ssp.getPoiGroupById('firstPoiGroup');
```

js

getPoiGroupByName

通过 `name` 查找 `poi` 组

定义:

```
function getPoiGroupByName(name: string): Group[];
```

ts

用法:

```
const groupList = ssp.getPoiGroupByName('name_firstPoiGroup');
```

js

removePoiGroupById

定义:

```
function removePoiGroupById(id: GroupInfo['id']): boolean;
```

ts

用法:

```
const isRemoveSuccess = ssp.removePoiGroupById('firstPoiGroup');
```

js

clearPoi

清除当前场景内所有 `poi` 对象。

定义:

```
function clearPoi(): void;
```

ts

用法:

```
ssp.clearPoi();
```

js

showAllPoi

显示当前场景内所有 `poi` 对象。

定义:

用法：

```
ssp.showAllPoi();
```

js

hideAllPoi

隐藏当前场景内所有 `poi` 对象。

定义：

```
function hideAllPoi(): void;
```

ts

用法：

```
ssp.hideAllPoi();
```

js

[← 通用模型](#)[PoiNode 对象 →](#)