

plugin-soonmanager2-sync

npm@latest v2.9.10

SoonManager2.0 平台 [🔗](#) 生产的场景加载及数据读取。

注意

此插件为 CPS 平台旧版资源使用，新版资源包请使用 `cps-soonmanager` 插件

样例



播放动画

停止动画

显示拓扑路径

删除拓扑路径

Close Controls

<>

安装

```
npm install @soonspacejs/plugin-soonmanager2-sync -S  
# or
```

sh

使用方法

```
import SoonSpace from 'soonspacejs';
import Soonmanager2SyncPlugin from '@soonspacejs/plugin-soonmanager2-sync';

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

const soonmanager2Sync = ssp.registerPlugin(
  Soonmanager2SyncPlugin,
  'soonmanager2Sync'
);

console.log(soonmanager2Sync);
```

属性

path

资源加载的基础路径

- 默认值: ''
- 类型: string

metaData

场景元数据

- 默认值: null
- 类型: IMetadata | null

☰ SoonSpace.js 2.x

定义

```
interface IMetadata {  
  platformVersion: number;  
  version: number;  
  name: string;  
  projectId: string;  
  sceneId: string;  
  cover: string | null;  
  flatModel: string;  
  treeModel: string;  
  exportTime: number;  
}
```

ts

treeData

场景树数据

- 默认值: `null`
- 类型: `ITreeData[] | null`

定义

```
interface ITreeData {  
  id: string;  
  pid: string | null;  
  name: string;  
  renderType: 'GROUP' | '3D' | 'ROOM' | 'STUB';  
  matrix: number[];  
  path: string | null;  
  children: ITreeData[];  
}
```

ts

topologyData

拓扑路径数据

☰ SoonSpace.js 2.x

propertiesData

自定义属性数据，根据 `modelId` 分组

- 默认值: `null`
- 类型: `TPropertiesMap | null`

定义

```
interface IProperties {  
  modelId: string;  
  group: string;  
  key: string;  
  value: string | null;  
  label: string | null;  
}  
  
type TPropertiesMap = Map<IProperties['modelId'], IProperties[]>;
```

ts

animationsData

补间动画数据，根据 `modelId` 分组

- 默认值: `null`
- 类型: `TAnimationsMap | null`

定义

```
interface IKeyframe {  
  id: string;  
  uuid: string;  
  x: number;  
  y: number;  
  z: number;  
  scaleX: number;  
  scaleY: number;  
  scaleZ: number;
```

ts

☰ SoonSpace.js 2.x

```
rotation2: number;  
easing: AnimationModeType;  
mode: string;  
delay: number;  
duration: number;  
repeat: number;  
yoyo: boolean;  
}  
  
/**  
 * 动画  
 */  
interface IAnimations {  
  id: string;  
  uuid: string;  
  modelId: string;  
  name: string;  
  keyframes: IKeyframe[];  
}  
  
type TAnimationsMap = Map<IAnimations['modelId'], IAnimations[]>;
```

modelVisionsData

模型视角数据，根据 `nodeId` 分组

- 默认值: `null`
- 类型: `TModelVisionsMap | null`

定义

```
interface IModelVisions {  
  id: string;  
  uuid: string;  
  nodeId: string;  
  name: string;  
  code?: any;  
  position: IVector3;  
  rotation: IVector3;  
  target: IVector3;
```

ts

☰ SoonSpace.js 2.x

```
type InodeEvisionSnap = Map<InodeEvisions[ InodeId ], InodeEvisions>;
```

方法

setPath

设置加载资源的基础路径

定义

```
function setPath(path: string): void;
```

ts

用法

```
soonmanager2Sync.setPath('./models');  
// or  
soonmanager2Sync.setPath('http://xxx.com/models');
```

js

注意

插件的其他方法依赖于 `path`，需要先设置才能使用

loadScene

加载场景对象

定义

```
interface ILoadSceneOptions {  
  /**  
   * 同步自定义属性  
   */
```

ts

☰ SoonSpace.js 2.x

```
    * 同步模型视图数据
    */
    syncModelVisions?: boolean;
    /**
     * 计算 bounds tree
     */
    needsModelsBoundsTree?: boolean;
}

function loadScene(options?: ILoadSceneOptions): Promise<void>;
```

用法

```
soonmanager2Sync.loadScene().then(() => {
  console.log('场景对象加载完成');
});
```

js

提示

如果你需要使用 Worker 计算 BVH，可以关闭默认行为

```
soonmanager2Sync.loadScene({ needsModelsBoundsTree: false }).then(() => {
  ssp.computeModelsBoundsTree({
    type: 'worker',
    workerCreator,
  });
});
```

具体请查看 [computeModelsBoundsTree](#)

参数

options

- 描述: 场景加载选项
- 必填: **✗**

SoonSpace.js 2.x

ILoadSceneOptions

属性	描述	类型	必填	默认值
syncProperties	是否同步自定义属性	boolean	✗	true
syncModelVisions	是否同步节点视角数据	boolean	✗	true
needsModelsBoundsTree	场景加载完成后调用 computeModelsBoundsTree 方法	boolean	✗	true

提示

自定义属性存储在对象的 `userData.properties` 属性上

getTopologies

获取拓扑路径数据

定义

```
function getTopologies(): Promise<TopologyInfo[]>;
```

ts

用法

```
soonmanager2Sync.getTopologies().then((topologies) => {  
  const [t1] = topologies;  
  
  /**  
   * 使用获取到的数据直接创建拓扑路径  
   */  
  ssp.createTopology(t1);  
});
```

js

☰ SoonSpace.js 2.x

playAnimationById

根据对象 `id` 播放补间动画

定义

```
type TAnimationsTweenProps = Pick<
  IKeyframe,
  | 'x'
  | 'y'
  | 'z'
  | 'rotationX'
  | 'rotationY'
  | 'rotationZ'
  | 'scaleX'
  | 'scaleY'
  | 'scaleZ'
>;

interface IPlayAnimationByIdOptions {
  onUpdate?: (
    source: TAnimationsTweenProps,
    tween: Tween<TAnimationsTweenProps>
  ) => void;
  onStart?: (tween: Tween<TAnimationsTweenProps>) => void;
}

function playAnimationById(
  id: string,
  animationIndex?: number,
  options?: IPlayAnimationByIdOptions
): Promise<void>;
```

用法

```
soonmanager2SyncPlugin.playAnimationById('4H6T1H53CSFW', 0, {
  onUpdate: (source, tween) => {},
  onStart: (tween) => {
    /**
```

☰

SoonSpace.js 2.x

```
        console.log(tween);  
    },  
    });  
});
```

参数

id

- 描述: 要播放动画的对象 `id`
- 必填: ✓
- 类型: `string`

animationIndex

- 描述: 该动画所在数据列表中的下标
- 必填: ✗
- 默认值: `0`
- 类型: `number`

options

- 描述: 动画播放选项
- 必填: ✗
- 类型: `IPlayAnimationByIdOptions`

IPlayAnimationByIdOptions

属性	描述	类型	必填	默认值
<code>onUpdate</code>	动画更新回调	<code>IPlayAnimationByIdOptions['onUpdate']</code>	✗	
<code>onStart</code>	动画开始回调	<code>IPlayAnimationByIdOptions['onStart']</code>	✗	

提示

动画播放时, 可以是多个 `animation` 的组合

☰ SoonSpace.js 2.x

fetchMetaData

根据当前 `path` 获取场景元数据

定义

```
function fetchMetaData(): Promise<IMetadata>;
```

ts

用法

```
soonmanager2SyncPlugin.fetchMetaData().then((metaData) => {  
  console.log(fetchMetaData);  
});
```

ts

fetchTreeData

根据当前 `path` 获取场景树数据

定义

```
function fetchTreeData(): Promise<ITreeData[]>;
```

ts

用法

```
soonmanager2SyncPlugin.fetchTreeData().then((treeData) => {  
  console.log(treeData);  
});
```

ts

提示

☰ SoonSpace.js 2.x

fetchPropertiesData

根据当前 `path` 获取自定义属性数据

定义

```
function fetchPropertiesData(): Promise<TPropertiesMap>;
```

ts

用法

```
soonmanager2SyncPlugin.fetchPropertiesData().then((propertiesData) => {  
  console.log(propertiesData);  
});
```

ts

fetchAnimationsData

根据当前 `path` 获取补间动画数据

定义

```
function fetchAnimationsData(): Promise<TAnimationsMap>;
```

ts

用法

```
soonmanager2SyncPlugin.fetchAnimationsData().then((animationsData) => {  
  console.log(animationsData);  
});
```

ts

fetchModelVisionsData

根据当前 `path` 获取模型视角数据

☰ SoonSpace.js 2.x

定义

```
function fetchModelVisionsData(): Promise<TModelVisionsMap>;
```

ts

用法

```
soonmanager2SyncPlugin.fetchModelVisionsData().then((modelVisions) => {  
  console.log(modelVisions);  
});
```

ts

[← plugin-soonmanager-sync](#)[plugin-cps-soonmanager →](#)