

答题卡组件

答题卡组件包含普通答题和快速问答两种类型的组件，两种组件采用同样的事件体系，所以仅需要实例化一个答题卡SDK，将同一个实例传给两种不同的答题组件使用。

- 普通答题有单选、多选、评分和投票这几种题型，并且还具有限时答题功能。
- 快速问答界面与普通答题不同，且仅包含单选和多选题目。

在引入及使用本功能组件前，需要先对互动功能接收端SDK进行公共配置，详情请查看：[互动功能接收端 UI 组件 - 配置 SDK](#)。

普通答题

引入

在线文件引入方式

```
// script 标签引入，根据版本号引入JS版本。
<script src="https://websdk.videocc.net/interactions-receive-sdk-ui-default/0.24.0/lib/PcAnswerCard/PcAnswerCard.umd.min.js"></script>
// PC端
<script>
    const AnswerCard = window.PolyvIRScene.PcAnswerCard.default;
</script>
```

```
// script 标签引入，根据版本号引入JS版本。
<script src="https://websdk.videocc.net/interactions-receive-sdk-ui-default/0.24.0/lib/MobileAnswerCard/MobileAnswerCard.umd.min.js"></script>
// 移动端
<script>
    const AnswerCard = window.PolyvIRScene.MobileAnswerCard.default;
</script>
```

import 方式引入（推荐）

```
// PC端
import AnswerCard from '@polyv/interactions-receive-sdk-ui-default/lib/PcAnswerCard/index';
```

```
// 移动端
import AnswerCard from '@polyv/interactions-receive-sdk-ui-default/lib/MobileAnswerCard/index';
```

使用方式

答题卡组件内部基于 SDK 处理了大部分业务逻辑，接入方仅需监听 `to-show` 和 `to-hide` 事件，去控制组件的可见性即可。另外，`status-changed` 事件会在答题数据发生变化时触发，并且含有答题状态，可根据该状态做相应处理，比如修改模态框/提示文字等。

代码示例

以 PC 端接入为例：

```

<template>
  <div v-show="isShowAnswerCard">
    <h2>{{ answerCardTitle }}</h2>
    <answer-card
      :answer-card-sdk="answerCardSdk"
      @to-show="onSetAnswerCardVisible(true)"
      @to-hide="onSetAnswerCardVisible(false)"
      @status-changed="onStatusChanged"
    />
  </div>
</template>

<script>
import { AnswerCard as AnswerCardSDK } from '@polyv/interactions-receive-sdk';
import AnswerCard from '@polyv/interactions-receive-sdk-ui-default/lib/PcAnswerCard/index';

export default {
  components: {
    AnswerCard,
  },

  data() {
    return {
      // 答题卡SDK实例
      answerCardSdk,
      // 是否显示答题卡
      isShowAnswerCard: false,
      // 外部弹窗（容器）标题
      answerCardTitle: '',
    };
  },

  created() {
    this.answerCardSdk = new AnswerCardSDK();
  },

  beforeDestroy() {
    this.answerCardSdk?.destroy();
  },

  methods: {
    onSetAnswerCardVisible(visible) {
      this.isShowAnswerCard = visible;
    },

    onStatusChanged(status) {
      if (status === 'isShowResult' || status === 'isShowAnswer') {
        this.answerCardTitle = this.answerCardSdk.curSubmittedAnswer ? '答题结果' : '未作答';
      } else {
        this.answerCardTitle = '答题卡';
      }
    },
  },
},

```

```
};
</script>
```

Attributes

属性名	类型	默认值	含义
answerCardSdk	Object	null	答题卡 SDK 实例
lang	string: 'zh_CN', 'en'	'zh_CN'	语言
delayTime	number	移动端 8000，PC端 2000	延迟触发显示时间，单位毫秒

Events

事件名	参数类型	参数含义	说明
to-show	Object	题目数据	收到题目后触发，表示需要展示弹窗
to-hide	Object	题目数据	答题终止后触发，表示需要隐藏弹窗
status-changed	string: 'isShowQuestion', 'isShowResult', 'isShowAnswer'	答题卡内部展示状态	答题卡内部展示状态发生变化时触发。 isShowQuestion：正在显示题目，PC和移动端组件共有状态； isShowResult：正在显示答题结果；PC和移动端组件共有状态； isShowAnswer：正在显示答案，移动端组件特有状态。可根据此状态，执行修改弹窗标题等操作。见demo示范。

to-show 事件参数说明

to-show 事件包含一个对象类型的事件参数，该对象具体字段如下。

属性名	类型	含义
itemType	number: 0, 1	答题卡类型, 0 表示普通答题卡, 1 表示快速问答
type	string: 'C', 'R', 'S', 'V'	题目类型, C: 多选题, R: 单选题, S: 评分题, V: 投票题
name	string	题目描述
questionId	string	题目id
duration	number	限时题目特有字段, 答题时间, 单位秒
options	string[]	选项数组
tips	string[]	提示数组 (评分题特有, 分值提示)

to-hide 事件参数说明

to-hide 事件包含一个对象类型的事件参数, 该对象具体字段如下。

属性名	类型	含义
itemType	number: 0, 1	答题卡类型, 0 表示普通答题卡, 1 表示快速问答
type	string	题目类型
questionId	string	题目id

Methods

方法名	参数	说明
backToResult	无参数	将答题卡组件内部状态切换成"isShowResult" (查看结果) 状态, 仅适用于移动端组件

快速问答

接入逻辑同普通答题。可基于 `to-show` 和 `to-hide` 事件的参数判断题目为单选/多选。

引入

在线文件引入方式

```
// script 标签引入，根据版本号引入JS版本。
<script src="https://websdk.videocc.net/interactions-receive-sdk-ui-default/0.24.0/lib/PcQuickAnswer/PcQuickAnswer.umd.min.js"></script>
// PC端
const QuickAnswerCard = window.PolyvIRScene.PcQuickAnswer.default;
```

```
// script 标签引入，根据版本号引入JS版本。
<script src="https://websdk.videocc.net/interactions-receive-sdk-ui-default/0.24.0/lib/MobileQuickAnswer/MobileQuickAnswer.umd.min.js"></script>
// 移动端
const QuickAnswerCard = window.PolyvIRScene.MobileQuickAnswer.default;
```

import 方式引入（推荐）

```
// PC 端
import QuickAnswerCard from '@polyv/interactions-receive-sdk-ui-default/lib/PcQuickAnswer/index';
```

```
// 移动端
import QuickAnswerCard from '@polyv/interactions-receive-sdk-ui-default/lib/MobileQuickAnswer/index';
```

使用方式

同样地，PC端快速问答仅提供内容元素，所以需要接入方设置容器元素和控制可见性。

代码示例

```
// 快速问答基础 mixin
import { AnswerCard as AnswerCardSDK } from '@polyv/interactions-receive-sdk';

export default {
  data() {
    return {
      // 答题卡SDK实例
      answerCardSdk,
      // 是否显示答题卡
      isShowAnswerCard: false,
    };
  },

  created() {
    this.answerCardSdk = new AnswerCardSDK();
  },

  beforeDestroy() {
    this.answerCardSdk?.destroy();
  },

  methods: {
    onSetQuickAnswerCardVisible(visible, question) {
      this.isShowAnswerCard = visible;
    },
  },
};
```

```
<!-- @file 快速问答 PC 端 -->
<template>
  <div v-show="isShowAnswerCard">
    <quick-answer-card
      :answer-card-sdk="answerCardSdk"
      @to-show="(question) => onSetQuickAnswerCardVisible(true, question)"
      @to-hide="(question) => onSetQuickAnswerCardVisible(false, question)"
    />
  </div>
</template>

<script>
import mixin from './mixin';

export default {
  mixins: [mixin],
};
</script>
```

而web移动端展示形式为横向长条形的答题卡，目前Polyv SaaS 平台直播观看页会将其展示在移动端菜单之上。移动端组件内部会控制快速问答的显示与否。

```
<!-- @file 快速问答移动端 -->
<template>
  <div>
    <quick-answer-card
      :answer-card-sdk="answerCardSdk"
    />
  </div>
</template>

<script>
import mixin from './mixin';

export default {
  mixins: [mixin],
};
</script>
```

Attributes

属性名	类型	默认值	含义
answerCardSdk	Object	null	答题卡 SDK 实例

Events

事件名	参数类型	参数含义	说明
to-show	无参数	-	收到题目后触发，表示需要展示组件
to-hide	无参数	-	答题终止后触发，表示需要隐藏组件
success-submit	无参数	-	提交答题卡成功后回调