

红包组件

目前针对特定客户开放使用。

本文档主要描述如何接入互动功能接收端的红包组件，接入后可实现观众端领取普通红包、口令红包和红包雨的三种领取类型红包。

积分红包： 中奖后，红包会自动发放到用户的积分账户中，用户可以在积分记录中查看领取积分历史。

支付宝红包： 中奖后，可获得支付宝口令，复制该口令，可在支付宝中搜索领取红包。

注意：本功能目前只支持0.11.1-beta.2及以上版本,如要使用请安装对应版本。

```
npm i @polyv/interactions-receive-sdk@0.15.7
```

```
npm i @polyv/interactions-receive-sdk-ui-default@0.15.7
```

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](#)。

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.videooc.net/interactions-receive-sdk-ui-default/0.24.0/lib/MobileRedEnvelope/MobileRedEnvelope.umd.min.js"></script>  
// 移动端  
<script>  
  const MobileRedEnvelope = window.PolyvIRScene.MobileRedEnvelope.default;  
</script>
```

import 方式引入（推荐）

```
import { MobileRedEnvelope } from '@polyv/interactions-receive-sdk-ui-default/lib/PcWelfareLottery';
```

组件注入-红包实例 代码示例

```
import { RedEnvelope } from '@polyv/interactions-receive-sdk';
export default {
  // 注入 redEnvelopeSdk 给子组件使用, redEnvelopeSdk 是红包实例 只实例化一次
  provide() {
    return {
      redEnvelopeSdk: new RedEnvelope()
    };
  },
  // 业务逻辑
  ...
}
```

以移动端接入为例红包组件：

```

<template>
  <div>
    <red-envelope
      :red-envelope-sdk="redEnvelopeSdk"
      :redEnvelopeData="redEnvelopeData"
      :coverImg="redPackSkin"
      :openId="openId"
      :lang="lang"
      :withDrawEnabled="withDrawEnabled"
      :redirect-url="watchCodeUrl"
      :unit="donateSetting.scoreUnit"
      @copy-success="$emit('copySuccess')"
      @close="handleCloseRedEnvelope"
      @statusChange="handleStatusChange"
      @click-withdraw="handleClickWithdraw"
      @click-point-record="handleClickPoint"
      @received="handleReceived"
    />
  </div>
</template>

<script>
import RedEnvelope from '@polyv/interactions-receive-sdk-ui-default/lib/MobileRedEnvelope';
import mixin from './mixin';
export default {
  mixins: [mixin],
  components: {
    RedEnvelope
  },
  methods: {
    handleClickWithdraw() {
      // TODO 点击提现，跳转到提现页面
      this.redEnvelopeData = null;
    },
    handleReceived(pwd) {
      // TODO 红包领取成功， pwd 为口令红包口令，领取成功后需要发言
    },
    handleClickPoint() {
      // TODO 点击积分记录，展示积分记录弹窗
      this.redEnvelopeData = null;
    },
  },
}
};
</script>

```

```

/**
 * @file 红包组件逻辑
 */
import { mapState } from 'vuex';

export default {
  data() {
    return {
      // 红包信息
      redEnvelopeData: null,
      // 红包主体显示
      showRedEnvelope: false,
    };
  },
  inject: ['redEnvelopeSdk'],
  props: {
    redPackSkin: {
      type: String,
      default: ''
    },
    withDrawEnabled: {
      type: Boolean,
      default: true,
    },
    // 观看地址
    watchCodeUrl: {
      type: String,
      default: '',
    },
    // 打赏设置
    donateSetting: {
      type: Object,
      default: () => ({}),
    },
  },
  computed: {
    ...mapState({
      lang: state => state.lang.lang,
    }),
    openId() {
      return this.redEnvelopeData?.user?.openId || this.redEnvelopeData?.openId || '';
    },
  },
  mounted() {
    this.bindAppEvents();
  },
  beforeUnmount() {
    this.unbindAppEvents();
  },
  methods: {
    handleCloseRedEnvelope() {
      this.redEnvelopeData = null;
    },
  },

```

```

    },
    handleOpenRedEnvelop(data) {
      const {
        content: greeting,
        type, redCacheId, redpackId, user: {
          pic: avatar,
          nick: nickname
        }
      } = data;
      if (type.includes('rain')) {
        // 打开红包雨组件
        this.$eventBus.$emit('OPEN_REDPACK_RAIN', {
          redCacheId,
          redPackId: redpackId,
          avatar,
          nickname,
          greeting,
          type
        });
      } else {
        this.redEnvelopeData = data;
      }
    },
    handleStatusChange(redpackId, status) {
      // 通知聊天室窗口改变红包状态
      this.$eventBus.$emit('SET_ENTRANCE_STATUS', {
        redpackId,
        status
      });
    }
  },
};

```

Attributes

属性名	类型	默认值	含义
red-envelope-sdk	Object	null	红包 SDK 实例
redEnvelopeData	Object	null	外部传入红包数据
custom-lang-config	Object	null	中英文配置 默认中英文配置文件
custom-img-config	Object	null	图片配置
coverImg	String	"	红包封面

属性名	类型	默认值	含义
openId	String	"	微信openId 从获取 redEnvelopeData.user.openId
withDrawEnabled	Boolean	"	是否展示提现入口或者积分记录入口
redirectUrl	String	"	二维码自定义重定向url
unit	String	"	频道设置积分单位
lang	String: 'zh_CN', 'en'	'zh_CN'	语言

customImgConfig

属性名	类型	默认值	含义
normalEntranceImg	string	null	消息封面图-普通红包
passwordEntranceImg	string	null	消息封面图-口令红包
rainEntranceImg	string	null	消息封面图-红包雨红包
normalPendantImg	string	null	挂件图-普通红包
passwordPendantImg	string	null	挂件图-口令红包
rainPendantImg	string	null	挂件图-红包雨红包
redpackCoverImg	string	null	弹出封面图-普通红包和口令红包
rainModalTopBgImg	string	null	弹出层顶部背景图-红包雨
rainModalRightPendantImg	string	null	弹出层右侧挂件图-红包雨
redpackBeginingBgImg	string	null	红包雨开始倒计时图
rainThemeBottomBgImg	string	null	红包雨动画-底部背景图
rainThemeBollImgArray	array	null	红包雨动画-飘落图 (随机选择数组图片)

Event

事件名	回调参数	说明
copy-success	boolean	复制直播间链接成功回调
close	boolean	关闭红包弹窗回调
statusChange	红包id , 红包状态	redpackId: String; status: received 已领取, nonReceived 没抢到, joined 首次加入
click-withdraw	无	点击提现入口回调
click-point-record	无	点击积分记录回调
received	口令	pwd: string 领取口令红包后，需要发送口令消息

红包挂件组件 代码示例

```
<template>
  <red-envelop-pendant
    v-if="showPendant"
    :type="redEnvelopeDelayData.type || redEnvelopeDelayData.redpackType"
    :time="redEnvelopeDelayData.delayTime"
    @end="handleCloseRedPendant"
  />
</template>

<script>
import RedEnvelopPendant from '@polyv/interactions-receive-sdk-ui-default/lib/RedEnvelopePendant';
import mixin from './mixin';
export default {
  mixins: [mixin],
  components: {
    RedEnvelopPendant
  },
};
</script>
```

```

/**
 * @file 红包挂件组件逻辑
 */
export default {
  data() {
    return {
      // 红包挂件显示
      showPendant: false,
      // 倒计时红包内容
      redEnvelopeDelayData: null,
      // 接口是否请求完毕
      getNewestRequestEnd: true,
    };
  },
  watch: {
    redEnvelopeSdk: {
      immediate: true,
      handler() {
        this.handleRedPaper = this.handleRedPaperEvent.bind(this);
        this.handleRedPaperDelay = this.handleRedPaperDelayEvent.bind(this);
        this.bindAppEvents();
      }
    },
  },
  // 父组件注入 redEnvelopeSdk 实例
  inject: ['redEnvelopeSdk'],
  mounted() {
    // 是否有延时红包
    this.getNewest();
    this.handleVisibilitychange();
  },
  beforeUnmount() {
    this.unbindAppEvents();
  },

  methods: {
    // 是否有延时红包
    async getNewest() {
      this.getNewestRequestEnd = false;
      const result = await this.redEnvelopeSdk.getNewest();
      this.getNewestRequestEnd = true;
      const { code, data } = result;
      const delayTime = data ? Math.round((data.sendTime - data.serverTime) / 1000) : 0;
      if (code === 200 && data && data.timeEnabled === 'Y' && delayTime > 0) {
        this.redEnvelopeDelayData = {
          ...data,
          pic: data.avatar,
          delayTime
        };
        this.showPendant = true;
      }
    },
    // 监听红包相关事件
    bindAppEvents() {
      const redEnvelopeSdk = this.redEnvelopeSdk;

```

```

        redEnvelopeSdk && redEnvelopeSdk
            .on(redEnvelopeSdk.events.REDPAPER_FOR_DELAY, this.handleRedPaperDelay)
            .on(redEnvelopeSdk.events.REDPAPER, this.handleRedPaper);
    },
    // 取消监听红包相关事件
    unbindAppEvents() {
        const redEnvelopeSdk = this.redEnvelopeSdk;
        redEnvelopeSdk && redEnvelopeSdk
            .off(redEnvelopeSdk.events.REDPAPER_FOR_DELAY, this.handleRedPaperDelay)
            .off(redEnvelopeSdk.events.REDPAPER, this.handleRedPaper);
    },
    handleRedPaperDelayEvent(data) {
        this.redEnvelopeDelayData = data;
        this.showPendant = true;
    },
    handleRedPaperEvent() {
        this.showPendant = false;
    },
    handleCloseRedPendant() {
        this.showPendant = false;
    },
    handleVisibilitychange() {
        // 页面最近一次隐藏的时间
        let latestTime;
        document.addEventListener('visibilitychange', () => {
            if (document.visibilityState === 'visible' && this.getNewestRequestEnd && latestTime && (Date.now() -
latestTime) > 1000) {
                this.getNewest();
            }
            if (document.visibilityState === 'hidden') {
                latestTime = Date.now();
            }
        });
    },
};
```

Attributes

属性名	类型	默认值	含义
type	String	'normal'	根据红包类型展示封面
time	Number	0	传入时间戳
custom-lang-config	Object	null	中英文配置 默认中英文配置文件
custom-img-config	Object	null	图片配置

Event

事件名	回调参数	说明
end	无	倒计时完成

红包消息组件 代码示例

```
<!-- 聊天列表红包消息组件内容 -->
<template>
  <div class="c-red-paper-msg">
    <red-envelope-entrance
      ref="redEnvelopEntrance"
      :red-envelope-data="msg"
      :red-envelope-sdk="redEnvelopeSdk"
      :lang="lang"
      @open="handleRedpackVisible"/>
    </div>
  </template>

<script>
import RedEnvelopeEntrance from '@polyv/interactions-receive-sdk-ui-default/lib/RedEnvelopeEntrance';
import mixin from '../mixins';

export default {
  mixins: [mixin],

  components: {
    RedEnvelopeEntrance
  },

  props: {
    msg: Object,
  },
};
</script>
```

```
export default {
  inject: ['redEnvelopeSdk'],
  mounted() {
    this.$eventBus.$on('SET_ENTRANCE_STATUS', this.handleStatusUpdate);
  },
  beforeDestroy() {
    this.$eventBus.$off('SET_ENTRANCE_STATUS', this.handleStatusUpdate);
  },
  props: {
    msg: {
      type: Object
    },
    lang: {
      type: String
    },
  },
  methods: {
    handleRedpackVisible() {
      this.$eventBus.$emit('OPEN_REDPAPER', this.msg);
    },
    handleStatusUpdate({
      redpackId,
      status
    }) {
      if (this.msg.redpackId === redpackId) {
        // 更新消息列表红包状态
        this.$refs.redEnvelopEntrance.updateStatus(status);
      }
    },
  },
}
};
```

Attributes

属性名	类型	默认值	含义
red-envelope-sdk	Object	null	红包消息SDK 实例
red-envelope-data	Object	null	外部传入红包数据
lang	String: 'zh_CN', 'en'	'zh_CN'	语言
custom-lang-config	Object	null	中英文配置 默认中英文配置文件
custom-img-config	Object	null	图片配置

Event

事件名	回调参数	说明
open	无	点击回调

红包雨组件 代码示例

```
<template>
  <div>
    <redpack-rain-comp
      ref="redpackRain"
      :redpack-rain="redpackRainSdk"
      :lang="lang"
      :customThemeImgs="customRedpackThemeImgs"
      :withDrawEnabled="withDrawEnabled"
      :redirect-url="watchCodeUrl"
      :unit="donateSetting.scoreUnit"
      @click-withdraw="handleClickWithdraw"
      @copy-success="handleCopySuccess"
      @state-change="handleStateChange"
      @visible-change="handleVisibleChange"
      @click-point-record="handleClickPoint"
    />
  </div>
</template>

<script>
import mixin from './mixin';
import RedpackRainComp from '@polyv/interactions-receive-sdk-ui-default/lib/MobileRedpackRain';
export default {
  mixins: [mixin],
  components: {
    RedpackRainComp
  }
};
</script>
```

```

/**
 * @file 红包雨组件逻辑
 */
import { RedpackRain } from '@polyv/interactions-receive-sdk';

export default {
  props: {
    lang: {
      type: String,
      default: 'zh_CN'
    },
    customRedpackThemeImgs: {
      type: Object,
    },
    withDrawEnabled: {
      type: Boolean,
      default: true
    },
    // 观看地址
    watchCodeUrl: {
      type: String,
      default: '',
    },
    donateSetting: {
      type: Object,
      default: () => {
        return {};
      }
    },
  },
  data() {
    return {
      redpackRainSdk: null,
      redEnvelopeData: null
    };
  },
  created() {
    this.redpackRainSdk = new RedpackRain();
    // 监听消息列表红包雨点击事件
    this.$eventBus.$on('OPEN_REDPACK_RAIN', this.openRedpackRain);
  },
  beforeDestroy() {
    this.$eventBus.$off('OPEN_REDPACK_RAIN', this.openRedpackRain);
    this.redpackRainSdk = null;
  },
  methods: {
    // 处理点击提现
    handleClickWithdraw() {
    },
    // 处理点击积分记录
    handleClickPoint() {
    },
    // 处理复制成功
    handleCopySuccess() {

```

```
        // TODO Toast提示 复制成功
    },
    // 调用红包组件的展示方法
    openRedpackRain(data) {
        this.redEnvelopeData = data;
        this.$refs.redpackRain.showRedpackRain({
            ...data,
            source: 'msg'
        });
    },
    // 通知聊天室窗口改变红包状态
    handleStateChange({ redPackId, state }) {
        this.$eventBus.$emit('SET_ENTRANCE_STATUS', {
            redpackId: redPackId,
            status: state
        });
    },
    handleVisibleChange(visible) {
        // TODO 红包弹窗显示隐藏回调
    },
}
};
```

Attributes

属性名	类型	默认值	含义
redpack-rain	Object	null	红包雨 SDK 实例
customThemelmgs	String	"	红包封面
withDrawEnabled	Boolean	"	是否展示提现入口或者积分记录入口
redirect-url	String	"	二维码自定义重定向url
unit	String	"	频道设置积分单位
lang	String: 'zh_CN', 'en'	'zh_CN'	语言

Event

事件名	回调参数	说明
copy-success	boolean	复制直播间链接成功回调
close	boolean	关闭红包弹窗回调

事件名	回调参数	说明
state-change	红包id , 红包状态	redpackId: String; status: received 已领取, nonReceived 没抢到, joined 首次加入
click-withdraw	无	点击提现入口回调
click-point-record	无	点击积分记录回调
visible-change	boolean	红包弹窗显示隐藏回调

Methods

事件名	参数	说明
showRedpackRain	redEnvelopeData	红包数据

红包积分记录组件 代码示例

```
<template>
  <div class="plv-demo-point-re-record-default">
    <modal
      :visible="isShowResult"
      title="积分记录"
      :modal-style="{
        height: 'auto',
        position: 'absolute',
        bottom: '0',
      }"
      @close="isShowResult = false"
    >
      <MobilePointRERecord v-if="isShowResult" :red-envelope-sdk="redEnvelopeSdk" unit="点" />
    </modal>
  </div>
</template>

<script>
import MobilePointRERecord from '@polyv/interactions-receive-sdk-ui-default/lib/MobilePointRERecord';
// Modal 是移动端弹窗组件，
import Modal from '../MobileModal';

export default {
  components: {
    MobilePointRERecord,
    Modal
  },
  data() {
    return {
      // 红包 SDK 实例
      redEnvelopeSdk: null,
      isShowResult: true
    };
  },
  created() {
    this.redEnvelopeSdk = new RedEnvelope();
  },
};
</script>

<style lang="scss">
.plv-demo-point-re-record-default {
  width: 100%;
}
</style>
```

Attributes

属性名	类型	默认值	含义
red-envelope-sdk	Object	null	红包 SDK 实例
unit	string	"	积分单位