

# WebSocketSoftPhone 说明手册

BasicCall

提供簡單完美的呼叫中心產品



3.0

sp2017-006

2017-2-10

4008-110-512

[www.simperfect.com](http://www.simperfect.com)

文档版权属于北京信普飞科技有限公司

## 目 录

1. 概述.....	1
2. 系统结构 .....	2
2.1. 结构图 .....	2
2.2. WSSOFTPHONE.....	2
2.3. SOFTPHONEUI.....	2
2.4. 宿主页面与非宿主页面.....	3
3. WSSOFTPHONE 调用.....	4
3.1. 新手入门.....	4
3.1.1. DEMO 介绍.....	4
3.1.2. 软电话登录流程.....	4
3.1.3. 对象创建.....	5
3.1.4. 初始化参数 ATTRS.....	7
3.2. 高手进阶.....	8
3.2.1. 事件处理 EVENTHANDLER .....	8
3.2.2. 函数调用 .....	9
3.3. UI 界面的控制 .....	11
3.3.1. 功能按钮控制.....	11
3.3.2. 状态及提示信息显示.....	12
3.3.3. 队列信息 .....	12
3.3.4. 号码输入 .....	13
3.3.5. 座席列表 .....	14
4. 非宿主页面间接调用（跨域调用） .....	17
4.1. 外拨 .....	17
4.2. 转接 IVR.....	18
5. OCX 融合移植.....	19
5.1. 对象创建.....	19
5.2. 事件处理.....	19
5.3. 方法调用.....	19
5.4. 常见问题.....	20
5.4.1. WSSOFTPHONE 是否支持读取本地 INI 配置.....	20
5.4.2. 可否自定义 UI 界面.....	20
6. 系统特殊功能开关.....	21

6.1.	自定义工作模式.....	21
6.2.	离席模式，同步进入离席状态.....	21
6.3.	单步转移.....	22
6.4.	座席技能列表.....	22
6.5.	IVR 列表.....	23
6.6.	普通坐席及 RPN 座席功能屏蔽.....	23
7.	特殊功能应用.....	25
7.1.	重新加载数据库参数.....	25
7.2.	远端座席.....	25
7.3.	自定义技能及级别.....	25
7.4.	密码认证流程.....	25
7.5.	单步转移附带随路数据.....	26

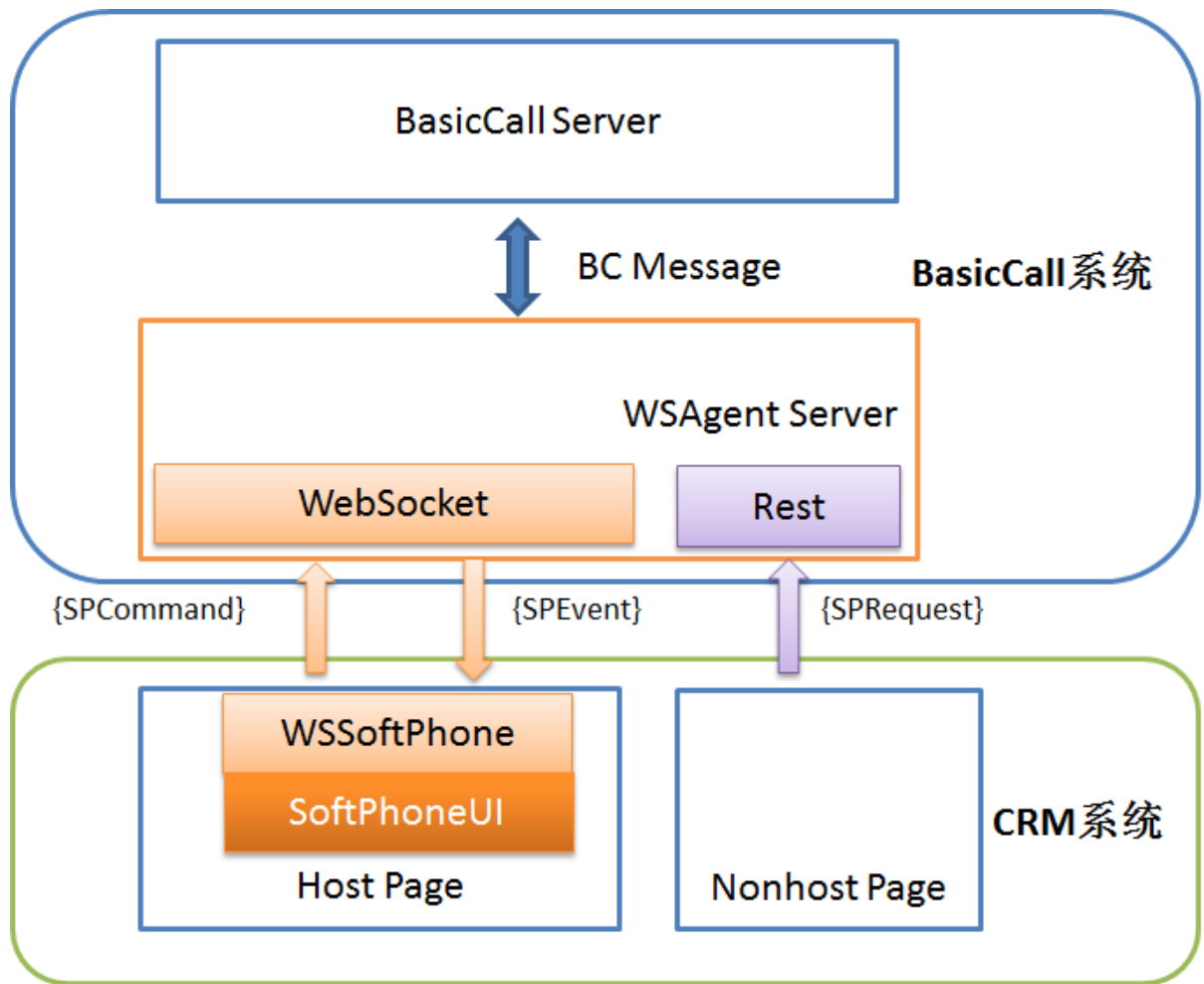
## 1. 概述

为了方便 Web 业务系统与软电话的融合，BasicCall 推出基于 WebSocket 的软电话模块。WebSocketSoftPhone 基于 HTML5 开发，要求座席使用的浏览器支持 HTML5，建议使用新版本的 Chrome 或者 FireFox。

CRM 中，SoftPhone 的宿主页面可以通过 js 创建软电话对象，并通过自定义的 UI 界面与 CTI 进行交互。同时系统 CRM 的其它非宿主页面通过 REST 请求，直接针对登录的软电话进行外拨、转接 IVR 等 CTI 操作。

## 2. 系统结构

### 2.1. 结构图



### 2.2. WSSoftPhone

WSSoftPhone 与 WSAgentServer 之间是通过 WebSocket 进行通讯的，实现软电话对象的创建、方法调用以及接收事件通知。

其它不包含 WSSoftPhone 对象的网页，则可以通过 Rest 请求调用软电话功能。

### 2.3. SoftphoneUI

为了方便使用，系统提供了基于 WSSoftPhone 封装的带界面的 SoftPhoneUI 对象，该对象基于 AmazeUI 框架，提供软电话交互界面。

13800138000 10010 系统测试用户	退出	普通 模式	结束通话	座席消息	外拨咨询	内部咨询	保持	监听
6002	通话	开始通话						

如果业务网站使用了其它的第三方 js 框架，可以参照 SoftPhoneUI 创建改造自定义风格

的 UI 交互界面。

#### 2.4. 宿主页面与非宿主页面

包含并创建软 WSSoftPhone 对象的页面为宿主页面，在其中的 js 可以直接访问 WSSoftPhone 对象，调用 CTI 功能。

非宿主页面为 CRM 系统中的其它页面，通常无法直接通过 js 访问到 WSSoftPhone 对象。在本架构中，可以通过 rest 接口来调用 CTI 功能。

### 3. WSSoftPhone 调用

#### 3.1. 新手入门

新手入门主要对软电话进行简单的介绍以及实现初步的登录功能的对接。

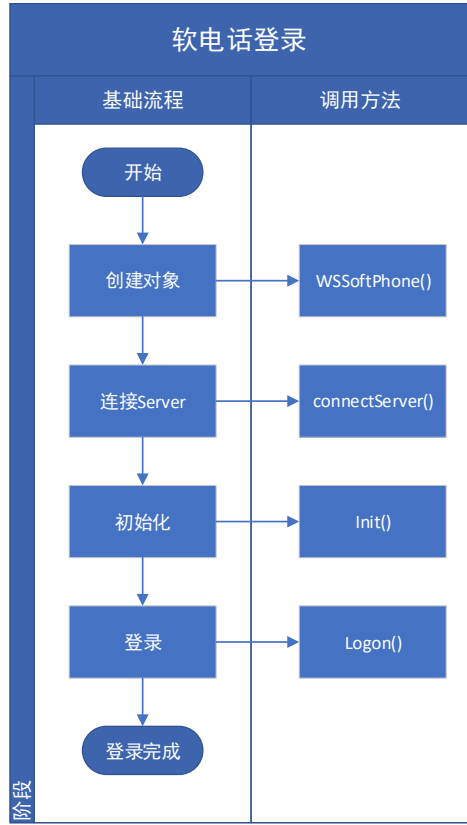
##### 3.1.1. Demo 介绍

为了方便对接，软电话会提供一份对接 demo 界面，此界面提供了对接样例，调用方式等。  
提供的 demo 在正式环境下正常使用。

名称	地址	主要作用
WSSoftPhoneDemo.html	SoftphoneDemo\htmlbootstrap3	软电话 demo 界面的操作入口
WSSoftPhoneDemo.js	SoftphoneDemo\htmlbootstrap3	软电话 demo 界面的操作入口调用的相应方法
SPCommand.js	SoftphoneDemo\js\wsSoftPhone	
WSSoftPhone.js	SoftphoneDemo\js\wsSoftPhone	软电话对象的大部分调用方法
SoftphoneUI.js	SoftphoneDemo\js\sdk	软电话 UI 对象的大部分调用方法

##### 3.1.2. 软电话登录流程

在 CRM 应用中，如何正确的对接并登录软电话，基础流程如下图所示：



3.1.3. 对象创建

在 CRM 应用中，建议创建且只创建一个 WSSoftPhone 对象。

对象定义：

**function** WSSoftPhone(attrs, eventHandler)

在创建时，可以初始化相关属性 **attrs**，并注入事件的处理函数 **eventHandler**。

范例：

```
if (!this.wssp) {
    this.wssp = new WSSoftPhone({
        agentId : this.settings.agentId,
        agentName : this.settings.agentName,
        ins : this.settings.devId,
        serverUri : this.settings.serverUri,
        delegatorId : this.settings.delegatorId,
        workModeSettings : this.settings.workModeSettings
    }, {
        onInited : function(delegatorId) {
            self.onInited(delegatorId);
        }
    });
}
```



```
    },
    onLogonSucc : function(ins) {
        self.onLogonSucc(ins);
    },
    onLogoffSucc : function(ins) {
        self.onLogoffSucc(ins);
    },
    onTrace : function(strmsg) {
        self.trace(strmsg);
    },
    onStateChange : function(state) {
        self.onStateChange(state);
    },
    onWorkModeChange : function(workmode) {
        self.onWorkModeChange(workmode);
    },
    onQueueChange : function(queueList) {
        self.onQueueChange(queueList);
    },
    onShowTip : function(info) {
        self.onShowTip(info);
    },
    onCallArrive : function(coInfos) {
        // 进行弹屏处理
        self.onCallArrive(coInfos);
    },
    onPrepareDialSucc : function() {
        // 进行号码输入界面或者通讯录显示
        self.onPrepareDialSucc();
    },
    onPrepareCallOutSucc : function() {
        // 显示号码输入界面或者通讯录显示
        self.onPrepareCallOutSucc();
    },
    onDialBegin : function(coInfos) {
        // 进行外拨信息显示
        self.onDialBegin(coInfos);
    },
    },
```

```
onDialSucc : function() {  
    // 进行外拨成功的业务逻辑操作  
    self.trace("do some CRM operation for DialSucc");  
},  
onDialError : function(reason) {  
    // 进行外拨失败的业务逻辑操作  
    self.trace("do some CRM operation for DialFailed");  
},  
onWebSocketReplaced : function() {  
    // 屏蔽软电话界面，仅显示重新创建的按钮  
    self.onWebSocketReplaced();  
},  
onConfirmReplaceWebSocket : function() {  
    // 存在已有连接，确认是否需要顶替  
    self.onConfirmReplaceWebSocket();  
},  
onRefreshAgentListSucc : function(agentList) {  
    // 显示座席列表供选择  
    self.onRefreshAgentListSucc(agentList);  
}  
  
});  
this.wssp.connectServer();  
self.trace("Create Succ!");  
  
}
```

3.1.4. 初始化参数 attrs

属性	功能	说明
agentId	座席工号	该工号必须在平台的数据库中存在，否则系统不运行登录
agentName	座席姓名	
ins	座席实例	一般为使用的分机号
serverUri	WebSocketServer 地址	默认为 8887 端口
delegatorId	代理 ID	该值一般用于刷新页面时恢复之前的软电话状态使用

workModeSettings	工作模式设定	如果系统使用特殊定义的工作模式，可以通过该参数初始化自定义的工作信息。
------------------	--------	-------------------------------------

### 3.2. 高手进阶

高手进阶主要讲解软电话的功能及基本函数的调用方式。软电话基本功能一览：

功能名称	作用	权限
坐席登录	登录功能，登录后软电话开始生效	All
坐席离席	离席功能，离席后不能进行电话操作	All
坐席外拨	坐席对外拨号并通话	All
班长监听	管理员可以监听正在通话的坐席	班长、管理员
班长控制	管理员可以强制签出或监听坐席	班长、管理员
坐席内呼	坐席与坐席之间的通话	All
外部咨询	当前坐席 A 和客户 B 通话，点击【外部咨询】后坐席 A 和被咨询者 C 通话，客户 B 处于保持状态。	All
内部咨询	当前坐席 A 和客户 B 通话，点击【内部咨询】后坐席 A 和被咨询坐席 C 通话，客户 B 处于保持状态。	All
电话保持	坐席通话中暂停与客户通话，客户电话处于被保持状态	All
电话转接	坐席通话中将电话转接给其他人	All
满意度调查	坐席通话中客户会进入到满意度调查的 IVR 流程，坐席直接结束通话	All

#### 3.2.1. 事件处理 eventHandler

在创建软电话时，可以传入事件处理的方法。

	事件	功能	说明
UI 控制	onInitd	系统初始化	通常在此保存 delegatorId，同时刷新 UI 界面
	onStateChange	状态改变	在此更新 UI 的功能按钮界面以及提示信息
	onWorkModeChange	工作模式改变	在此更新 UI 的功能按钮界面以及提示信息
	onQueueChange	队列变化	用于更新排队信息
输入选择	onPrepareDialSucc	外拨准备成功	在此显示外拨号码输入界面

	onPrepareCallOutSucc	外部咨询准备成功	在此显示外部咨询号码输入界面
	onRefreshAgentListSucc	座席列表刷新成功	显示可供选择的座席列表
软电话顶替	onWebSocketReplaced	软电话顶替	软电话被顶替，更新 UI，屏蔽功能操作
	onConfirmReplaceWebSocket	软电话顶替确认	提示是否进行软电话顶替操作
业务逻辑	onCallArrive	来电振铃	在此处理弹屏逻辑，获取呼叫信息并显示相关内容
	onDevConnected	通话开始	仅呼入时存在该事件，在此处理通话的逻辑，例如开始填写工单信息等
	onDialBegin	外拨开始	提示外拨开始
	onDialSucc	外拨成功	在此处理外拨成功的逻辑，显示对应外拨工单等
	onDialError	外拨失败	在此进行外拨失败的业务逻辑
提示信息	onLogonSucc	签入成功	进行提示
	onLogonFailed	签入失败	进行提示，errNum 及 errDesc 为相关的错误信息
	onLogoffSucc	签出成功	进行提示
	onShowTip	提示信息	显示提示信息
	onTrace	跟踪信息	用于显示跟踪信息

### 3.2.2. 函数调用

WSSoftPhone 对外提供如下函数可供使用

	事件	功能	说明
综合函数	connectServer	连接 Server	serverUri 为可选参数，一般不设置，使用初始化时使用的参数。
	callSPFunction	调用软电话功能	funcId 为 ESPFunc 的枚举值，对应各个软电话按钮功能
辅助函数	getCurStateName	获取当前状态名称	
	getCurWorkModeName	获取当前工作模式名称	

	getStateSetting	获取当前状态功能按钮设置	
	getCoInfos	获取当前呼叫信息对象	
	getQueueList	获取当前队列对象	
独立 CTI 函数	Logon	签入	
	Logoff	签出	
	setWorkMode	设置工作模式	wkmd 为工作模式的值
	Leave	离席	
	resumeWork	复席	
	Hold	保持	
	fetchHold	取保持	
	prepareDial	外拨准备	
	cancelPrepareDial	取消外拨准备	
	Dial	外拨	phoneNum 为外拨的电话号码，需要附带出局码
	cancelDial	取消外拨	
	dropCall	挂断	
	prepareConsult	内部咨询准备	
	Consult	内部咨询	destSPTtype: 目标类型，AGT 座席，EXT 分机 destSPIIns: 座席实例或者分机号
	cancelConsult	取消咨询	
	finishConsult	结束咨询	
	consultTransfer	咨询转接	
	prepareCallout	外部咨询准备	
	Callout	外部咨询	phoneNum 为外部咨询的电话号码，需要附带出局码
	prepareCallInner	内呼准备	
	cancelPrepareCallInner	取消内呼准备	

	refreshCallInnerAgentList	刷新内呼列表	
	callInner	内呼	destSPTType: 目标类型, AGT 座席, EXT 分机 destSPIns: 座席实例或者分机号
	cancelCallInner	取消内呼	
	prepareMonitor	监听准备	
	cancelPrepareMonitor	取消监听准备	
	refreshMonitorAgentList	刷新监听座席	
	monitorAgent	监听	destSPTType: 目标类型, AGT 座席, EXT 分机 destSPIns: 座席实例或者分机号
	finishMonitor	结束监听	
	consultConf	咨询三方会议	
	consultSwap	咨询切换	
	consultSwapBack	咨询切换	
	dropConf	结束三方会议	
	oneStepTransfer	单步转移	destType: 目标类型, AGT 座席, EXT 分机, OUT 外线 destIns: 分机号或外线号码
	transferToIVR	转接 IVR	

### 3.3. UI 界面的控制

#### 3.3.1. 功能按钮控制

软电话的功能按钮, 主要依靠处理 onStateChange 事件来进行控制。在 onStateChange 回调时, 通过 wssp.getStateSetting() 来获取系统建议的按钮设置, 同时改变 UI 界面。



```
var setting = this.wssp.getStateSetting();
if (setting) {
    var html = "<div><a>State: ' + setting.state + '</a>";
    for (var i = 0; i < setting.functions.length; i++) {
        var func = setting.functions[i];
```

```

    if (func.id == 36) {
        html += this.genWorkModeButton(func);
    } else {
        html += '<input type="button" onclick="' + this.objectName
            + '.spfunction(' + func.id + ')' value="'
            + getWSSoftPhoneText("SPFunc_" + func.id)
            + '" class="spbutton-' + func.id + ' button';

        if (!func.enabled) {
            html += ' disabled="disabled"';
        } else {
            html += ' white';
        }

        html += '>';
    }
}

html += '</div>';

this.divcontainer.find(".softphone .spbuttons").html(html);

```

### 3.3.2. 状态及提示信息显示

为了使座席能够很好地掌握当前的 CTI 信息，建议在 UI 界面中能够显示当前的状态，以及系统提供的提示信息。

当前的状态信息是在 onStateChange 时，获取 getCurStateName 进行更新。

提示信息是在接收到 onShowTip 时，显示对应的提示信息。

6002	空闲	登录成功
------	----	------

### 3.3.3. 队列信息

队列信息用于使座席掌握当前的排队情况，在 onQueueChange 事件中处理传递过来的 queueList 对象来进行显示。

13800138000	1001	0	系统测试用户
13800138000	1001	0	系统测试用户

对象格式如下：

```

[
  {
    "mCOType": "",

```

```
"level": 0,
"cotype": 0,
"ani": "13800138000",
"aid": "系统测试用户",
"isOut": "",
"dnis": "10000",
"coid": "BC_1_58aaabff_4001",
"sk": "1001",
"qtm": 0,
"reserved": null,
"tenantID": "",
"isValid": true,
"keyStr": "",
"iconKey": "",
"phoneNum": "13800138000"
},
{
  "mCOType": "",
  "level": 0,
  "cotype": 0,
  "ani": "13800138000",
  "aid": "系统测试用户",
  "isOut": "",
  "dnis": "10000",
  "coid": "BC_3_58aaac14_4003",
  "sk": "1001",
  "qtm": 0,
  "reserved": null,
  "tenantID": "",
  "isValid": true,
  "keyStr": "",
  "iconKey": "",
  "phoneNum": "13800138000"
}
]
```

#### 3.3.4. 号码输入

在业务系统发起外拨时，一般通过点击业务上的电话号码，直接调用 dial 接口发起外拨。



另外一种方式就是点击软电话 UI 上的外拨按钮，手动输入号码来发起外拨。

此时需要调用 `prepareDial` 方法，并在 `onPrepareDialSucc` 事件时显示用户输入的模式窗口，让用户进行输入。SoftPhoneUI 中使用了 AmazeUI 的脚本封装来实现类似功能

```
SoftPhoneUI.prototype.onPrepareDialSucc = function() {  
    var self = this;  
    editForm.showEdit("请输入号码", "号码输入", "", function(options) {  
        trace("phoneNum=" + options.data);  
        self.wssp.dial(options.data);  
    }, function() {  
        trace("canceled!");  
        self.wssp.cancelPrepareDial();  
    });  
}
```



### 3.3.5. 座席列表

系统在内呼、咨询等操作时，需要弹出座席列表供座席选择。WSSoftPhone 会在 `onRefreshAgentListSucc` 事件回调中将座席列表数据传送过来。UI 需要利用合适的方式进行展示，建议采用 `modal` 的方式进行展示。座席必须选择或者取消后再做下一步处理。

传递的对象信息如下：

```
{  
  "agentListType": "AGENTLISTTYPE_CALLINNER",  
  "items": [  
    {  
      "state": "3",  
      "itemType": "1",  
      "ins": "6002",  
      "workID": "888",  
      "agentName": "员工 888",  
    }  
  ]  
}
```

```
        "workMode": "0",
        "leaveReason": null,
        "dept": "1",
        "extnum": null,
        "devowner": null,
        "calst": null,
        "dworkID": null
    },
    {
        "state": null,
        "itemType": "0",
        "ins": "6001",
        "workID": null,
        "agentName": "",
        "workMode": null,
        "leaveReason": null,
        "dept": null,
        "extnum": "6001",
        "devowner": null,
        "calst": "32",
        "dworkID": null
    },
    {
        "state": null,
        "itemType": "0",
        "ins": "6004",
        "workID": null,
        "agentName": "",
        "workMode": null,
        "leaveReason": null,
        "dept": null,
        "extnum": "6004",
        "devowner": null,
        "calst": "32",
        "dworkID": null
    }
  ]
}
```

选择坐席

☒ 坐席列表 ☐ 分机列表

设备	名称	工号	状态	
<input type="radio"/>	6003	测试工号	999	空闲-普通模式

取消

提交

#### 4. 非宿主页面间接调用（跨域调用）

页面除了通过直接访问 WSSoftPhone 对象，直接调用函数外，还可以调用 WSAgentServer 的接口来实现 CTI 功能。接入定义如下：

**public** SPResponse processCommand(String jsonCommand)

其中 jsonCommand 为 SPCommand 对象的 json 字符串。

其它页面在调用前，需要通过 session 或者其它方法，获取到目标 WSSoftPhone 的 ins 和 delegatorId，然后组织 SPCommand 对象，进行 Ajax 调用。

##### 4.1. 外拨

外拨方法如下：

```
function serviceDial() {
    trace("serviceDial...");
    var delegatorId = $("#delegatorId").val(); // 后期代理 ID
    var ins = $("#ins").val(); // 获取座席实例
    var phoneNum = $("#txt_phoneNum").val(); // 获取外拨的号码
    var ecommand = ESPCommand.Dial; // 设定命令为外拨
    var cmd = new SPCommand(null, ecommand, ins, delegatorId);
    cmd.pushParams("phoneNum", phoneNum); // 增加参数
    var strJson = JSON.stringify(cmd); // 生成 json 字符串
    // ajax 调用 WSAgentServer 方法
    $.ajax({
        // 实际 CRM 融合时，需要将 url 替换为 WSAgentServer 的实际地址
        url: "../softphone/ processJsonpCommand.do ",
        dataType: 'jsonp',
        crossDomain: true,
        data: {
            jsonCommand : strJson
        },
        success: function(data) {
            trace("dial return:" + data.errorCode);
        }
    });
}
```

## 4.2. 转接 IVR

转接方法如下：

```
function serviceTransferIVR() {  
    trace("serviceTransferIVR....");  
    var delegatorId = $("#delegatorId").val(); // 后期代理 ID  
    var ins = $("#ins").val(); // 获取座席实例  
    var ecommand = ESPCommand.TransferToIVR; // 设定命令为转接 IVR  
    var cmd = new SPCommand(null, ecommand, ins, delegatorId);  
    cmd.pushParams("SVN", "999"); // 接入码  
    cmd.pushParams("TRANSFERPREPARE_CHECKPASSWORD", "0"); // 非密码验证  
    cmd.pushParams("OrderID", "SVC_1234567"); // 其它业务属性  
  
    var strJson = JSON.stringify(cmd); // 生成 json 字符串  
    // ajax 调用 WSAgentServer 方法  
    $.ajax({  
        url: "../softphone/ processJsonpCommand.do ",  
        dataType: 'jsonp',  
        data: {  
            jsonCommand: strJson  
        },  
        success: function(data) {  
            trace("dial return:" + data.errorCode);  
        }  
    });  
}
```

## 5. OCX 融合移植

原来进行 OCX 融合的业务，移植到 WebSocket 版本后，主要的差别点如下：

### 5.1. 对象创建

对象不再依赖 Object 标签，而使用纯 JS 的 WebSocket 通讯来实现，要求浏览器支持 WebSocket 通讯，建议使用高版本的 Chrome\FireFox 及 IE。

WSSoftPhone 登录的座席工号，要求在 CTI 平台存在，同时座席的呼叫权限等以 CTI 平台的配置为准。

原来 OCX 中 setInitParam 设置的很多参数将不再需要，统一以 CTI 平台的设置为准。

### 5.2. 事件处理

原来针对 Object 对象的事件订阅，需要调整为在创建 WSSoftPhone 或者 SoftPhoneUI 对象的事件处理声明。

功能	Object 事件定义	WSSoftPhone 事件回调
签入成功	evtLogonSucc	onLogonSucc
签入失败	evtLogonFailed	onLogonFailed
签出成功	evtLogoffSucc	onLogoffSucc
状态改变	evtStateChange	onStateChange
外拨开始	evtDialBegin	onDialBegin
外拨成功	evtDialSucc	onDialSucc
外拨失败	evtDialFailed	onDialFailed
来电振铃	evtCallArrive	onCallArrive
来电电话	evtDevConnected	onDevConnected

### 5.3. 方法调用

功能	方法	WSSoftPhone 方法
获取呼叫属性	getCOInfo	调用 getCoInfos 方法获取呼叫对象，直接访问各个属性即可。

获取座席属性	getAgentInfo	调用 getAgentInfos 方法获取座席信息对象，直接访问各个属性即可
设置呼叫信息	setCOInfo	调用 setCOInfo 方法
转接 IVR	transferToIVR	调用 transferToIVR 方法，传入 SPTransferIVRInfo 参数
外拨	Dial	调用 dial 方法
外部咨询	callout	调用 callout 方法

## 5.4. 常见问题

### 5.4.1. WSSoftPhone 是否支持读取本地 ini 配置

答：不支持。

WSSoftPhone 为纯 js 对方封装，无法读取原来的坐席本地配置 AgentPortal.ini 文件。

在初始化 WSSoftPhone 时至少需要确认两个坐席参数：坐席工号 and 使用的分机号。  
建议在 CRM 登录页面中，除了输入工号密码外，增加一个使用分机号的显示及配置的地方。同时利用 localStorage，存储上一次配置的分机号，减少坐席的工作量。

例如：

```
var ins = localStorage.getItem("WSDemo.ins");
if (ins) {
    $("#ins").val(ins);
}
```

保存代码

```
var ins = $("#ins").val();
localStorage.setItem("WSDemo.ins", ins);
```

### 5.4.2. 可否自定义 UI 界面

答：CRM 可以根据自己的美工效果自定义 UI 界面。

目前系统默认提供的 SoftphoneUI 是基于 AmazeUI 定义的 UI 界面。CRM 可以根据自身使用的 JS 框架及标签，替换掉 SoftPhoneUI 中使用到的页面元素，实现自定义的 UI 界面。

## 6. 系统特殊功能开关

### 6.1. 自定义工作模式

在系统 AgentWorkMode 表中对相应的模式进行设置。

ServiceWorkMode	ServiceWorkModeName	BasicWorkMode	Visible	BindSkillCode	SortOrder
-1	空 模式	-1	0		0
0	示闲	0	1		1
1	下班 模式	1	0		6
2	非实时 模式	2	0		5
3	示忙	3	1		3
4	班长 模式	4	0		2
5	预览 模式	5	0		4
6	离开	6	1		7
7	闭塞 模式	7	0		8

上例中，系统仅启用了三种模式可供座席直接选择：

其中将允许接听电话的“普通模式”显示为“示闲”；

主要从事外拨工作的“外拨模式”显示为“示忙”；

不接听电话的“离席模式”显示为“离开”。

### 6.2. 离席模式，同步进入离席状态

系统 SystemParam 表设置

ParamKey	ParamValue	取值说明
SOFTPHONELEAVEWORKMODESTATESYNC	1	1 代表启用离席模式、离席状态同步。

设置生效后，点击[离开]工作模式，系统会自动进入离席状态。

退出	离开	外拨	结束通话	座席消息	内呼	复席	监听
RP_9622	离席	离席成功					

从离席模式变化到[空闲]模式，系统会自动从离席状态，恢复到空闲状态。

退出	示闲	结束通话	座席消息	外拨	内呼	离席	监听
RP_9622	空闲	复席成功					



6.3. 单步转移

系统 SystemParam 表设置

ParamKey	ParamValue	取值说明
ONESTEPTRANSFERSUPPORT	1	1 代表启用，在通话时可以选择转接菜单。



6.4. 座席技能列表

系统 SystemParam 表设置

ParamKey	ParamValue	取值说明
AGENTTRANSFERSKILLINFO	1002,(中文)紧急问题 1003,(英语)英语服务	技能之间用 分割

选择转接技能

技能	名称
<input type="radio"/>	1002 (中文)紧急问题
<input checked="" type="radio"/>	1003 (英语)英语服务

取消

提交

6.5. IVR 列表

ParamKey	ParamValue	取值说明
AGENTTRANSFERIVRINFO	999,满意度 59006258,分销商主 流程 8888,验证密码,1	流程之间用 分割。 密码认证流程存在第三个参 数为 1.

选择IVR

接入码	流程名称	密码认证
<input type="radio"/>	999 满意度	false
<input type="radio"/>	59006258 分销商主流程	false
<input type="radio"/>	8888 验证密码	true

取消

提交

6.6. 普通坐席及 RPN 座席功能屏蔽

普通坐席的功能屏蔽可以通过设置 SystemParam 来实现

ParamKey	ParamValue	取值说明
----------	------------	------

SOFTPHONEMASKENABLED	33,34	功能之间用逗号分隔 对应的值为按钮的功能 ID， 33 为离席，34 为复席
----------------------	-------	----------------------------------------------

RPN 座席由于采用普通固话或手机来作为座席话机，功能上会有不小限制。默认需要屏蔽如下功能 (4,5,6,7,8,42,25,26,27) ， 可以通过设置 SystemParam 设置 SOFTPHONEMASKENABLED\_RPN

ParamKey	ParamValue	取值说明
SOFTPHONEMASKENABLED_RPN	4,5,6,7,8,42,25,26,27	功能之间用逗号分隔 对应的值为按钮的功能 ID

退出	示闲	结束通话	座席消息	外拨	内呼	离席	监听
RP_9622	空闲	登录成功					

## 7. 特殊功能应用

### 7.1. 重新加载数据库参数

BasicCall 中会在数据库里存放很多功能特性的配置参数，WSAgentServer 在启动时会默认加载。如果运行过程中，系统需要调整修改参数，例如修改“软电话可以直接转接的 IVR 流程”设置，那么此时需要手动触发 WSAgentServer 重新加载数据库配置参数，方面如下：

`http://<WebServer:Port>/WSAgentServer/softphone/reloadSystemParam.do`

正常情况下，Server 会返回 true，代表加载成功。

然后再打开网页，重新登录软电话进行测试。

### 7.2. 远端座席

在创建软电话时，请确认接听的电话号码（附带出局码），并将座席的 devId 设置带“RP\_”前缀的信息。例如：

```
settings.devId = "RP_" + $("#phoneNum").val();
```

### 7.3. 自定义技能及级别

如果业务系统希望根据 CRM 系统的信息来决定座席的技能及参数，可以通过在初始化时设定 customInitParams 参数。

初始化 SoftPhoneUI 时该参数为 SoftPhoneUISetting 中的属性；初始化 WSSoftPhone 则为初始化 Attrs 中的属性。

```
// 设置自定义参数
var initParams = new CustomInitParms();
initParams.agentWrapupSeconds = 10;
initParams.addSkill("1001", "5");
initParams.addSkill("1002", "1");
settings.customInitParams = initParams;
```

目前可以自定义的参数包括自动话后完成时间，以及技能和对应的技能级别。

（技能参数默认使用数据库配置参数。如果设置了自定义技能参数，则以自定义技能参数为准。）

### 7.4. 密码认证流程

