
Ver.	asterCC-1.2-05121
Date:	2013-05-12

asterCC Commercial API Manual (1.2)

Written by asterCC

Table of contents

1	Getting Start.....	5
1.1	About this Manual.....	5
2	Introduction	6
2.1	Overview.....	6
3	WebService API.....	7
3.1	Agent Login.....	8
3.2	Agent Logout	9
3.3	Agent group (queue) login/logout	10
3.4	Agent group (queue) Pause/Unpause	11
3.5	ACW (After Call Work) Mode Config.....	12
3.6	Exit ACW Mode	13
3.7	Workway Mode Config.....	14
3.8	Originate Call	15
3.9	Call Consult.....	16
3.10	Call Transfer.....	17
3.11	End a Call Consult.....	18
3.12	Conference.....	19
3.13	Hangup	20
3.14	Barge-in	21
3.15	Spy.....	22
3.16	Agent Status	23
3.17	Whisper.....	24
3.18	Agents Group Stauts.....	25
3.19	Call Release	26
3.20	Hold a Call	27
3.21	Unhold a Call	28
3.22	Predictive Dialer.....	29
4	HTTP API.....	30
4.1	Agent Login.....	31
4.2	Agent Logout	32
4.3	Agent group (queue) login/logout	33
4.4	Agent group (queue) Pause/Unpause	34
4.5	ACW (After Call Work) Mode Config.....	35

4.6	Exit ACW Mode	36
4.7	Workway Mode Config.....	37
4.8	Originate Call	38
4.9	Call Consult	39
4.10	Call Transfer	40
4.11	End a Call Consult.....	41
4.12	Conference.....	42
4.13	Hangup	43
4.14	Barge-in	44
4.15	Spy.....	45
4.16	Agent Status	46
4.17	Whisper.....	47
4.18	Agents Group Stauts.....	48
4.19	Call Release	49
4.20	Hold a Call	50
4.21	Unhold a Call	51
4.22	Predictive Dialer	52
5	JavaScript API	53
5.1	Agent Login.....	54
5.2	Agent Logout	56
5.3	Agent group (queue) login/logout	57
5.4	Agent group (queue) Pause/Unpause	58
5.5	ACW (After Call Work) Mode Config.....	59
5.6	Exit ACW Mode	60
5.7	Workway Mode Config.....	61
5.8	Originate Call	62
5.9	Call Consult	63
5.10	Call Transfer	64
5.11	End a Call Consult.....	65
5.12	Conference.....	66
5.13	Hangup	67
5.14	Barge-in	68
5.15	Spy.....	69
5.16	Agent Status	70
5.17	Whisper.....	71

5.18	Agents Group Stauts.....	72
5.19	Call Release	73
5.20	Hold a Call	74
5.21	Unhold a Call	75
5.22	Predictive Dialer	76
6	Receive events via javascript	77
6.1	Dial out:	78
6.1.1	Agent ring:.....	78
6.1.2	Agent answer:	78
6.1.3	Customer ring:	78
6.1.4	Customer answer:.....	79
6.1.5	Customer Hangup:.....	79
6.1.6	Agent hangup:.....	79
6.1.7	Conversation hangup:.....	79
6.2	Dial in:.....	80
6.2.1	Customer dialin:	80
6.2.2	Agent ring:.....	80
6.2.3	Agent answer:	80
6.2.4	Customer hangup:	80
6.2.5	Agent hangup:.....	81
6.2.6	Conversation hangup:.....	81
7	Receive events via http post.....	83
8	Appendix	85
8.1	Language Code	85
8.2	Error Codes	86

1 Getting Start

1.1 About this Manual

This manual is written for the developers who want to integrate asterCC with their own system. With the integration, developers can add simple features like click-call, call popup or create a full-featured agent interface without using asterCC agent portal page.

asterCC could work with both B/S or C/S application.

For better understanding, readers are better download and install asterCC Commercial system first, which can be downloaded from <http://astercc.org/downloads>.

Some useful links are list here, you'd better read these first:

Installation: http://wiki.astercc.com/doku.php?id=en:download_and_install:installation

Quick Start: http://wiki.astercc.com/doku.php?id=en:newbie:quick_start

asterCC Events:

http://wiki.astercc.com/doku.php?id=en:custom_development_guide:how_to_use_http_p_ussh_sample_page_to_receive_system_events

asterCC API:

http://wiki.astercc.com/doku.php?id=en:custom_development_guide:how_to_use_js_api_sample_page

Customer Page:

http://wiki.astercc.com/doku.php?id=en:custom_development_guide:how_to_integrate_custom_page_in_astercc

2 Introduction

2.1 Overview

To integrate a third party system, asterCC provides three API, developers could use either of these as they like.

I Web Service

I HTTP

I JavaScript

3 Webservice API

asterCC provides a web service to receive the requests, then it will call the core to finish the function and return the result. Webservice API includes agent control and call control.

Web service Address: [http://\\$\(ipaddress\):\\$\(port\)/\\$\(appname\)/](http://$(ipaddress):$(port)/$(appname)/)

Introduction:

\$(ipaddress) — IP address of the astercc server

\$(port) — the port of the HTTP service (it is 80 by default)

\$(appname) — the name of the application (it is supposed to be astcccwi.wsdl)

There are two files in app/webroot: astcccwi.php and astcccwi.wsdl

You would change the IP address in the bottom of astcccwi.wsdl when you need

```
552
553 <service name="astccCWI">
554     <port name="astccCWIPort" binding="tns:astccCWIBinding">
555         <soap:address location="http://192.168.1.45:4580/astcccwi.php"/>
556     </port>
557 </service>
```

For example, for user login, this is the php code to call asterCC webservice API

```
1 <?php
2     error_reporting(0);
3     header("Content-Type: text/html; charset=utf-8");
4     ini_set("soap.wsdl_cache_enabled", "0");
5     $client = new SoapClient("http://192.168.1.45:4580/astcccwi.php?wsdl");
6     try(
7         $backmsg = $client->login('orgidentity', 'usertype', 'user', 'pwdtype', 'password');
8         echo $backmsg;
9     )catch(SoapFault $fault){
10        echo "Fault! code:", $fault->faultcode, " string: ", $fault->faultstring;
11    }
12
13    exit;
14 ?>
```

3.1 Agent Login

Function:

login(orgidentity, usertype, user, pwdtype, password)

Description:

Send a login request to asterCC;

Parameters:

Param	Type	Description
orgidentity	String	The identity string of the team
usertype	String	User Type: agent/account
user	String	Use Agent No. when user type is agent Use Username when user type is account
pwdtype	String	Password Type: plaintext/MD5
password	String	Use plaintext or md5 password for agent or account

Return Format:

|Return|code|Return|message|Return|param

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message Return param	Succeeded Messages Parmas
Return 2 Return message	Failed Messages

3.2 Agent Logout

Function:

logout(orgidentity, usertype, user, pwdtype, password)

Description:

Send a logout request to asterCC;

Parameters:

Param	Type	Description
orgidentity	String	The identity string of the team
usertype	String	User Type: agent/account
user	String	Use Agent No. when user type is agent Use Username when user type is account
pwdtype	String	Password Type: plaintext/MD5
password	String	Use plaintext or md5 password for agent or account

Return Format:

|Return|code|Return|message|Return|param

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

3.3 Agent group (queue) login/logout

Function:

queueAction(type,user,orgidentity,list, pwdtype,password)

Description:

Agent login for one or several agents group

Parameters:

Params	Type	Description
type	String	1(login) / 2(logout)
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team
list	String	Put a string combine with agentgroup id, end with a comma, if you don't put the list, it will login/logout from all agent group the agent belongs to
pwdtype	String	Password Type: plaintext/MD5
password	String	Use plaintext or md5 password for agent or account

A valid list format example: agentgroupid1,agentgroupid2,.....agentgroupidn,

Return Format:

|Return|code|Return|message

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

3.4 Agent group (queue) Pause/Unpause

Function:

queuePause(type,usertype,user,orgidentity, pwdtype,password,paused_reason)

Description:

Pause/Unpause the agent (only when he is login)

Parameters:

Params	Type	Description
type	String	1(Pause)/2(Unpause)
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team
pwdtype	String	Password Type: plaintext/MD5
password	String	password
paused_reason	String	Put a pause reason when it's for pause Use the following string, if you need statistics training / meeting / leave / lunch / rest / other

Return Format:

|Return|code|Return|message

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

3.5 ACW (After Call Work) Mode Config

Function:

acwAction(type,usertype,user,orgidentity, pwdtype, password, agent_group_id)

Description:

Change the ACW mode of the agent in agent group

Parameters:

Params	Type	Description
type	String	1(Ringing ACW) / 2(Answered ACW) / 3(Disable ACW)
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
agent_group_id	String	ids of the agent group

ACW Mode, when agent is in ACW mode, agent would not receive any calls from queue.

- n** Ringing ACW: As long as the agent phone rings, no matter if agent answer, when the call hangup, agent will be in ACW mode
- n** Answered ACW: Only the call is answered, agent will be in ACW mode when call hangup.
- n** Disable ACW: agent would not go to ACW mode automaticly

Return Format:

|Return|code|Return|message

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

3.6 Exit ACW Mode

Function:

acwOff(sessionid,user,orgidentity, pwdtype,password)

Description:

Exit the ACW Mode if agent is in

Parameters:

Params	Type	Description
sessionid	String	Session id of the current call (you can get this from asterCC events)
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password

Return Format:

|Return|code|Return|message

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

3.7 Workway Mode Config

Function:

workwayAction(status,user,orgidentity, pwdtype,password,agent_group_id)

Description:

Change the agent work mode in agent group

Parameters:

Params	Type	Description
status	String	dialin (Dial in only) / dialout (Dial out only) / all
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
agent_group_id	String	ids of the agent group
callbackFuc	String	call back function

Return Format:

|Return|code|Return|message

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

3.8 Originate Call

Function:

makeCall(targetdn, targettype, agentgroupid, usertype, user, orgidentity, pwdtype, password, modeltype, model_id, userdata)

Description:

Originate a call

Parameters:

Params	Type	Description
targetdn	String	Agent No. (if it's a internal call, and they must belong to same team) or put phone number if it's an external call
targettype	String	exter (external call) / inner (internal call)
agentgroupid	String	id of the agent group
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
modeltype	String	BusinessApp / Campaign / Virtualcustomer / Customerservice
model_id	String	id of the model which you want to originate the call for
userdata	String	user data, only valid when modeltype= Campaign, say you put "customer_id=109&pwd=asdfg" here, when you receive the call events, you will get userdata=' customer_id=109&pwd=asdfg'

Return Format:

|Return|code|Return|message

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

3.9 Call Consult

Function:

consult(targetdn, agentgroupid, consulttype, agentuniqueid, calleruniqueid, pwdtype, password, usertype, user, orgidentity)

Description:

When agent in a call or in a conference, make a call to a new party, when it's answered, agent will talk to new party, the old party will be held(MOH) or stay in conference

Parameters:

Params	Type	Description
targetdn	String	Agent No. (internal consult) or phone number (external consult)
agentgroupid	String	agent group id when it's a internal consult, put 0 if it's a external consult
consulttype	String	internal / external
agentuniqueid	String	agent uniqueid of the agent
calleruniqueid	String	uniqueid of the caller
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team

Return Format:

|Return|code|Return|message

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

3.10 Call Transfer

Function:

transfer(sessionid, nwaycalluniqueid, calleruniqueid, agentuniqueid, pwdtype, password, usertype, user, orgidentity)

Description:

When call consult success, use call transfer will connect the consult and the call, and the agent will be hangup from the call.

Parameters:

Params	Type	Description
sessionid	String	sessionid of the current call
nwaycalluniqueid	String	uniqueid of the consult party
calleruniqueid	String	uniqueid of the caller
agentuniqueid	String	uniqueid of the agent
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team

Return Format:

|Return|code|Return|message

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

3.11 End a Call Consult

Function:

callReturn(nwaycalluniqueid, calleruniqueid, agentuniqueid, pwdtype, password, usertype, user, orgidentity)

Description:

When call consult success, agent hangup the consult party and go back to the call.

Parameters:

Params	Type	Description
nwaycalluniqueid	String	uniqueid of the consult party
calleruniqueid	String	uniqueid of the caller
agentuniqueid	String	uniqueid of the agent
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team

Return Format:

|Return|code|Return|message

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

3.12 Conference

Function:

conference(nwaycalluniqueid, calleruniqueid, agentuniqueid, consultcount, pwdtype, password, usertype, user, orgidentity)

Description:

When call consult success, connect the consult party to the call

Parameters:

Params	Type	Description
nwaycalluniqueid	String	uniqueid of the consult party
calleruniqueid	String	uniqueid of the caller
agentuniqueid	String	uniqueid of the agent
consultcount	String	how many parties in the conference besides caller and agent
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team

Return Format:

|Return|code|Return|message

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

3.13 Hangup

Function:

hangup(uniqueid, pwdtype, password, usertype, user, orgidentity)

Description:

Hangup one call party or a list of parties

Parameters:

Params	Type	Description
uniqueid	String	the uniqueid of the calls you want to hangup, end with a comma
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team

uniqueid example:

Hangup one party: uniqueid,

Hnagup: uniqueid1, uniqueid2,...uniqueidN,

Return Format:

|Return|code|Return|message

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

3.14 Barge-in

Function:

intrude(sessionid, phonenumber, pwdtype, password, usertype, user, orgidentity)

Description:

Agent Group admin could barge-in to a call

Parameters:

Params	Type	Description
sessionid	String	sessionid of the call
onenumber	String	the phone number you want to use for barge-in, you can use a extension in system or any phone number
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	passwd
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team

Return Format:

|Return|code|Return|message

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

3.15 Spy

Function:

silentMonitor(sessionid, phonenumber, pwdtype, password, usertype, user, orgidentity)

Description:

Agent Group admin could spy to a call

Parameters:

Params	Type	Description
sessionid	String	sessionid of the call
phonenumber	String	the phone number you want to use for spy, you can use a extension in system or any phone number
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team

Return Format:

|Return|code|Return|message

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

3.16 Agent Status

Function:

agentStatus(orgidentity, usertype, user, pwdtype, password)

Description:

Get agent status

Parameters:

Params	Type	Description
orgidentity	String	The identity string of the team
usertype	String	agent / account
user	String	Agent No. or Username
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password

Return Format:

|Return|code|Return|message|Return|status

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message Return status	Succeeded Messages Status String (when agent belongs to several agent group, it will use comma as separator)
Return 2 Return message	Failed Messages

format of statusstring:agentgroupid1-status1,agentgroupid2-status2,.....

agentgroupidn-statusn,

3.17 Whisper

Function:

whisper(sessionid,phonenumber,usertype,user, pwdtype,password,orgidentity)

Description:

Agent Group admin could talk to agent in a call, while all the other parties in the call can not hear

Parameters:

Params	Type	Description
sessionid	String	sessionid of the call
phonenumber	String	the phone number you want to use for whisper, you can use a extension in system or any phone number
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team

Return Format:

|Return|code|Return|message

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

3.18 Agents Group Status

Function:

```
agentgroupStatus(orgidentity,usertype,user,
  pwdtype,password,agent_group_id,agenttype)
```

Description:

Agent Group admin could get the status of the agents in his group

Parameters:

Params	Type	Description
orgidentity	String	The identity string of the team
usertype	String	agent / account
user	String	Agent No. or Username
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
agent_group_id	String	id of the agent group
agenttype	String	login(only check agents who is login) / all (all agents in the agent group)

Return Format:

|Return|code|Return|message

Type of the Return Value: String

Return Values:

Return	Description
Return 1 Return message Return status	Succeeded Messages Status String
Return 2 Return message	Failed Messages

Status String contains the status of one or several agents, each status will end with ||

for one agent: agent no,agent id,status||

for several agents:

agentno1-agentid1,status1||agentno2-agentid2,status2||.....agentnoN-agentidN,statusN||

Status: idle, ringing, busy, in use(agent have hangup, but the conversation is not, like caller is talking to a consult, but agent is free to make call), pause,logout

3.19 Call Release

Function:

forcedRelease(sessionid,phonenummer,usertype,user,pwdtype,password,orgidentity)

Description:

Agent group admin will hangup the agent and talk to caller

Parameters:

Params	Type	Description
sessionid	String	sessionid of the call
phonenummer	String	the phone number you want to use for call release, you can use a extension in system or any phone number
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team

Return Format:

|Return|code|Return|message|Return|status

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

3.20 Hold a Call

Function:

hold(agentuniqueid,calleruniqueid,orgidentity,usertype,user,pwdtype,password)

Description:

When it's two parties call, agent could put caller on hold, then caller will listen music

Parameters:

Params	Type	Description
agentuniqueid	String	call uniqueid of the agent
calleruniqueid	String	call uniqueid of the caller
orgidentity	String	The identity string of the team
usertype	String	agent / account
user	String	Agent No. or Username
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password

Return Format:

|Return|code|Return|message|Return|status

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

3.21 Unhold a Call

Function:

resume(agentuniqueid,calleruniqueid,orgidentity,usertype,user,pwdtype,password)

Description:

Get back the onhold call

Parameters:

Params	Type	Description
agentuniqueid	String	call uniqueid of the agent
calleruniqueid	String	call uniqueid of the caller
orgidentity	String	The identity string of the team
usertype	String	agent / account
user	String	Agent No. or Username
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password

Return Format:

|Return|code|Return|message|Return|status

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

3.22 Predictive Dialer

Function:

dialerlist (orgidentity, usertype, user, pwdtype, password, campaignid, phonenum, priority, dialtime)

Description:

Add phone numbers to the predictive dial list of a campaign

Parameters:

Params	Type	Description
orgidentity	String	The identity string of the team
usertype	String	agent / account
user	String	Agent No. or Username, the user must be group admin of the campaign
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
campaignid	String	id of the campaign
phonenum	String	phone numbers, use semicolon as the sperator when you have more than one. You can put 100 phone numbers at most, we suggest less than 50
priority	String	priority of the phone number in predictive dialer
dialtime	String	the dial time, if you give a empty string, it would be 0000-00-00 00:00:00

Return Format:

|Return|code|Return|message

Type of the Return Value:String

Return Values:

Value	Description
Return 1 Return message Return succ Return err	Succeeded Messages Number of Succeeded String of the failed numbers, use semicolon as the sperator
Return 2 Return message	Failed Messages

4 HTTP API

asterCC could also response for HTTP request. HTTP API could be used to get the agent status and control calls

HTTP URL: http://\$(ipaddress):\$(port)/asterccinterfaces

Description:

\$(ipaddress) — IP address of the astercc server

\$(port) — the port of the HTTP service (it is 80 by default)

For example, here is a php code for agent login

```

1  <?php
2  <!--
3  <code></code>
4
5  <code></code>
6  <code></code>
7  <code></code>
8
9  <code></code>
10 <code></code>
11 <code></code>
12 <code></code>
13 <code></code>
14 <code></code>
15 <code></code>
16 <code></code>
17 <code></code>
18 <code></code>
19 <code></code>
20 <code></code>
21 <code></code>
22 <code></code>
23 <code></code>
24 <code></code>
25 <code></code>
26 <code></code>
27 <code></code>

```

4.1 Agent Login

Reques String:

EVENT=LOGIN&**orgidentity**=orgidentity&**usertype**=usertype&**user**=user&**pwdtype**=pwdtype&**password**=password

Description:

Send a login request to asterCC;

Parameters:

Param	Type	Description
EVENT	string	LOGIN
orgidentity	String	The identity string of the team
usertype	String	User Type: agent/account
user	String	Use Agent No. when user type is agent Use Username when user type is account
pwdtype	String	Password Type: plaintext/MD5
password	String	Use plaintext or md5 password for agent or account

Return Format:

|Return|code|Return|message|Return|param

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message Return param	Succeeded Messages Parmas
Return 2 Return message	Failed Messages

4.2 Agent Logout

Reques String:

EVENT=LOGOUT&**orgidentity**=orgidentity&**usertype**=usertype&**user**=user&**pwdtype**=pwdtype&**password**=password

Description:

Send a logout request to asterCC;

Parameters:

Params	Type	Description
EVENT	String	LOGOUT
orgidentity	String	The identity string of the team
usertype	String	User Type: agent/account
user	String	Use Agent No. when user type is agent Use Username when user type is account
pwdtype	String	Password Type: plaintext/MD5
password	String	Use plaintext or md5 password for agent or account

Return Format:

|Return|code|Return|message

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

4.3 Agent group (queue) login/logout

Reques String:

EVENT=QUEUE&**type**=type&**usertype**=usertype&**user**=user&**orgidentity**=orgidentity&**list**
=list&**pwdtype**=pwdtype&**password**=password

Description:

Agent login for one or several agents group

Parameters:

Params	Type	Description
EVENT	String	QUEUE
type	String	1(login) / 2(logout)
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team
list	String	Put a string combine with agentgroup id, end with a comma, if you don't put the list, it will login/logout from all agent group the agent belongs to
pwdtype	String	Password Type: plaintext/MD5
password	String	Use plaintext or md5 password for agent or account

A valid list format example: agentgroupid1,agentgroupid2,.....agentgroupidn,

Return Format:

|Return|code|Return|message

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

4.4 Agent group (queue) Pause/Unpause

Reques String:

EVENT=QUEUEPAUSE&**type**=type&**usertype**=usertype&**user**=user&**orgidentity**=orgidentit
y&**pause_reason**=pause_reason&**pwdtype**=pwdtype&**password**=password

Description:

Pause/Unpause the agent (only when he is login)

Parameters:

Params	Type	Description
EVENT	String	QUEUEPAUSE
type	String	1(Pause)/2(Unpause)
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team
pwdtype	String	Password Type: plaintext/MD5
password	String	password
pause_reason	String	Put a pause reason when it's for pause Use the following string, if you need statistics training / meeting / leave / lunch / rest / other

Return Format:

|Return|code|Return|message

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

4.5 ACW (After Call Work) Mode Config

Reques String:

EVENT=ACW&type=type&usertype=usertype&user=user&orgidentity=orgidentity&agent_group_id=agent_group_id&pwdtype=pwdtype&password=password

Description:

Change the ACW mode of the agent in agent group

Params	Type	Description
EVENT	String	ACW
type	String	1(Ringing ACW) / 2(Answered ACW) / 3(Disable ACW)
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
agent_group_id	String	ids of the agent group

ACW Mode, when agent is in ACW mode, agent would not receive any calls from queue.

- n Ringing ACW: As long as the agent phone rings, no matter if agent answer, when the call hangup, agent will be in ACW mode
- n Answered ACW: Only the call is answered, agent will be in ACW mode when call hangup.
- n Disable ACW: agent would not go to ACW mode automaticly

Return Format:

|Return|code|Return|message

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

4.6 Exit ACW Mode

Reques String:

EVENT=ACWOFF&**sessionid**=sessionid&**usertype**=usertype&**user**=user&**orgidentity**=orgid
entity&**pwdtype**=pwdtype&**password**=password

Description:

Exit the ACW Mode if agent is in

Parameters:

Params	Type	Description
EVENT	String	ACWOFF
sessionid	String	Session id of the current call (you can get this from asterCC events)
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password

Return Format:

|Return|code|Return|message

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

4.7 Workway Mode Config

Reques String:

EVENT=WORKWAY&**status**=status&**usertype**=usertype&**user**=user&**orgidentity**=orgidentity&**pwdtype**=pwdtype&**password**=password&**agent_group_id**=agent_group_id

Description:

Change the agent work mode in agent group

Parameters:

Params	Type	Description
EVENT	String	WORKWAY
status	String	dialin (Dial in only) / dialout (Dial out only) / all
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
agent_group_id	String	ids of the agent group
callbackFuc	String	call back function

Return Format:

|Return|code|Return|message

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

4.8 Originate Call

Reques String:

EVENT=MAKECALL&targetdn=targetdn&targettype=targettype&agentgroupid=agentgroupid&usertype=usertype&user=user&orgidentity=orgidentity&pwdtype=pwdtype&password=password&modeltype=modeltype&model_id=model_id&userdata=userdata

Description:

Originate a call

Parameters:

Params	Type	Description
EVENT	String	MAKECALL
targetdn	String	Agent No. (if it's a internal call, and they must belong to same team) or put phone number if it's an external call
targettype	String	exter (external call) / inner (internal call)
agentgroupid	String	id of the agent group
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
modeltype	String	BusinessApp / Campaign / Virtualcustomer / Customerservice
model_id	String	id of the model which you want to originate the call for
userdata	String	user data, only valid when modeltype= Campaign, say you put "customer_id=109&pwd=asdfg" here, when you receive the call events, you will get userdata=' customer_id=109&pwd=asdfg'

Return Format:

|Return|code|Return|message

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

4.9 Call Consult

Reques String:

EVENT=CONSULT&targetdn=targetdn&agentgroupid=agentgroupid&consulttype=consult
type&agentuniqueid=agentuniqueid&calleruniqueid=calleruniqueid&pwdtype=pwdtype&p
assword=password&usertype=usertype&user=user&orgidentity=orgidentity

Description:

When agent in a call or in a conference, make a call to a new party, when it's answered, agent will talk to new party, the old party will be held(MOH) or stay in conference

Parameters:

Params	Type	Description
EVENT	String	CONSULT
targetdn	String	Agent No. (internal consult) or phone number (external consult)
agentgroupid	String	agent group id when it's a internal consult, put 0 if it's a external consult
consulttype	String	internal / external
agentuniqueid	String	agent uniqueid of the agent
calleruniqueid	String	uniqueid of the caller
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team

Return Format:

|Return|code|Return|message

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

4.10 Call Transfer

Reques String:

EVENT=TRANSFER&**sessionid**=sessionid&**nwaycalluniqueid**=nwaycalluniqueid&**calleruniqueid**=calleruniqueid&**agentuniqueid**=agentuniqueid&**pwdtype**=pwdtype&**password**=password&**usertype**=usertype&**user**=user&**orgidentity**=orgidentity

Description:

When call consult success, use call transfer will connect the consult and the call, and the agent will be hangup from the call.

Parameters:

Params	Type	Description
EVENT	String	TRANSFER
sessionid	String	sessionid of the current call
nwaycalluniqueid	String	uniqueid of the consult party
calleruniqueid	String	uniqueid of the caller
agentuniqueid	String	uniqueid of the agent
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team

Return Format:

|Return|code|Return|message

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

4.11 End a Call Consult

Reques String:

EVENT=CALLRETURN&**nwaycalluniqueid**=nwaycalluniqueid&**calleruniqueid**=calleruniqueid&**agentuniqueid**=agentuniqueid&**pwdtype**=pwdtype&**password**=password&**usertype**=usertype&**user**=user&**orgidentity**=orgidentity

Description:

When call consult success, agent hangup the consult party and go back to the call.

Parameters:

Params	Type	Description
EVENT	String	CALLRETURN
nwaycalluniqueid	String	uniqueid of the consult party
calleruniqueid	String	uniqueid of the caller
agentuniqueid	String	uniqueid of the agent
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team

Return Format:

|Return|code|Return|message

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

4.12 Conference

Reques String:

EVENT=CONFERENCE&nwaycalluniqueid=nwaycalluniqueid&calleruniqueid=calleruniqueid&agentuniqueid=agentuniqueid&consultcount=consultcount&pwdtype=pwdtype&password=password&usertype=usertype&user=user&orgidentity=orgidentity

Description:

When call consult success, connect the consult party to the call

Parameters:

Params	Type	Description
nwaycalluniqueid	String	uniqueid of the consult party
calleruniqueid	String	uniqueid of the caller
agentuniqueid	String	uniqueid of the agent
consultcount	String	how many parties in the conference besides caller and agent
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team

Return Format:

|Return|code|Return|message

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

4.13 Hangup

Reques String:

EVENT=HANGUP&**uniqueid**=uniqueid&**pwdtype**=pwdtype&**password**=password&**usertype**=usertype&**user**=user&**orgidentity**=orgidentity

Description:

Hangup one call party or a list of parties

Parameters:

Params	Type	Description
EVENT	String	HANGUP
uniqueid	String	the uniqueid of the calls you want to hangup, end with a comma
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team

uniqueid example:

Hangup one party: uniqueid,

Hnagup: uniqueid1, uniqueid2,...uniqueidN,

Return Format:

|Return|code|Return|message

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

4.14 Barge-in

Reques String:

EVENT=INTRUDE&**sessionid**=sessionid&**phonenumber**=phonenumber&**pwdtype**=pwdtype
 e&**password**=password&**usertype**=usertype&**user**=user&**orgidentity**=orgidentity

Description:

Agent Group admin could barge-in to a call

Parameters:

Params	Type	Description
EVENT	String	INTRUDE
sessionid	String	sessionid of the call
phonenumber	String	the phone number you want to use for barge-in, you can use a extension in system or any phone number
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	passwr
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team

Return Format:

|Return|code|Return|message

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

4.15 Spy

Reques String:

EVENT=SILENTMONITOR&**sessionid**=sessionid&**phonenumber**=phonenumber&**pwdtype**=pwdtype&**password**=password&**usertype**=usertype&**user**=user&**orgidentity**=orgidentity

Description:

Agent Group admin could spy to a call

Parameters:

Params	Type	Description
EVENT	String	SILENTMONITOR
sessionid	String	sessionid of the call
phonenumber	String	the phone number you want to use for spy, you can use a extension in system or any phone number
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team

Return Format:

|Return|code|Return|message

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

4.16 Agent Status

Reques String:

EVENT=AGENTSTATUS&**orgidentity**=orgidentity&**usertype**=usertype&**user**=user&**pwdtype**
e=pwdtype&**password**=password

Description:

Get agent status

Parameters:

Params	Type	Description
EVENT	String	AGENTSTATUS
orgidentity	String	The identity string of the team
usertype	String	agent / account
user	String	Agent No. or Username
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password

Return Format:

|Return|code|Return|message|Return|status

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message Return status	Succeeded Messages Status String (when agent belongs to several agent group, it will use comma as separator)
Return 2 Return message	Failed Messages

format of statusstring:agentgroupid1-status1,agentgroupid2-status2,.....

agentgroupidn-statusn,

4.17 Whisper

Reques String:

EVENT=Whisper&sessionid=sessionid&phonenumber=phonenumber&pwdtype=pwdtype
 &password=password&usertype=usertype&user=user&orgidentity=orgidentity

Description:

Agent Group admin could talk to agent in a call, while all the other parties in the call can not hear

Params	Type	Description
EVENT	String	Whisper
sessionid	String	sessionid of the call
phonenumber	String	the phone number you want to use for whisper, you can use a extension in system or any phone number
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team

Return Format:

|Return|code|Return|message

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

4.18 Agents Group Status

Request String:

EVENT=AgentgroupStatus&orgidentity=orgidentity&usertype=usertype&user=user&pwdtype=pwdtype&password=password&agent_group_id=agent_group_id&agenttype=agenttype

Agent Group admin could get the status of the agents in his group

Parameters:

Params	Type	Description
EVENT	String	AgentgroupStatus
orgidentity	String	The identity string of the team
usertype	String	agent / account
user	String	Agent No. or Username
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
agent_group_id	String	id of the agent group
agenttype	String	login(only check agents who is login) / all (all agents in the agent group)

Return Format:

|Return|code|Return|message

Type of the Return Value: String

Return Values:

Return	Description
Return 1 Return message Return status	Succeeded Messages Status String
Return 2 Return message	Failed Messages

Status String contains the status of one or several agents, each status will end with ||

for one agent: agent no,agent id,status||

for several agents:

agentno1-agentid1,status1||agentno2-agentid2,status2||.....agentnoN-agentidN,statusN||

Status: idle, ringing, busy, in use(agent have hangup, but the conversation is not, like caller is talking to a consult, but agent is free to make call), pause,logout

4.19 Call Release

Reques String:

EVENT=ForcedRelease&**sessionid**=sessionid&**phonenumber**=phonenumber&**pwdtype**=pwdtype&**password**=password&**usertype**=usertype&**user**=user&**orgidentity**=orgidentity

Description:

Agent group admin will hangup the agent and talk to caller

Parameters:

Params	Type	Description
EVENT	String	ForcedRelease
sessionid	String	sessionid of the call
phonenumber	String	the phone number you want to use for call release, you can use a extension in system or any phone number
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team

Return Format:

|Return|code|Return|message|Return|status

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

4.20 Hold a Call

Reques String:

EVENT=Hold&**agentuniqueid**=agentuniqueid&**calleruniqueid**=calleruniqueid&**orgidentity**=orgidentity&**usertype**=usertype&**user**=user&**pwdtype**=pwdtype&**password**=password

Description:

When it's two parties call, agent could put caller on hold, then caller will listen music

Parameters:

Params	Type	Description
EVENT	String	Hold
agentuniqueid	String	call uniqueid of the agent
calleruniqueid	String	call uniqueid of the caller
orgidentity	String	The identity string of the team
usertype	String	agent / account
user	String	Agent No. or Username
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password

Return Format:

|Return|code|Return|message|Return|status

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

4.21 Unhold a Call

Reques String:

EVENT=Resume&**agentuniqueid**=agentuniqueid&**calleruniqueid**=calleruniqueid&**orgidentity**=orgidentity&**usertype**=usertype&**user**=user&**pwdtype**=pwdtype&**password**=password

Description:

Get back the onhold call

Parameters:

Params	Type	Description
EVENT	String	Resume
agentuniqueid	String	call uniqueid of the agent
calleruniqueid	String	call uniqueid of the caller
orgidentity	String	The identity string of the team
usertype	String	agent / account
user	String	Agent No. or Username
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password

Return Format:

|Return|code|Return|message|Return|status

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

4.22 Predictive Dialer

Reques String:

EVENT=DialerList&**orgidentity**=orgidentity&**usertype**=usertype&**user**=user&**pwdtype**=pwdtype&**password**=password&**campaignid**=campaignid&**phonenum**=phonenum&**priority**=priority&**dialtime**=dialtime

Description:

Add phone numbers to the predictive dial list of a camaign

Parameters:

Params	Type	Description
EVENT	String	DialerList
orgidentity	String	The identity string of the team
usertype	String	agent / account
user	String	Agent No. or Username, the user must be group admin of the campaign
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
campaignid	String	id of the campaign
phonenum	String	phone numbers, use semicolon as the sperator when you have more than one. You can put 100 phone numbers at most, we suggest less than 50
priority	String	priority of the phone number in predictive dialer
dialtime	String	the dial time, if you give a empty string, it would be 0000-00-00 00:00:00

Return Format:

|Return|code|Return|message

Type of the Return Value:String

Return Values:

Value	Description
Return 1 Return message Return succ Return err	Succeeded Messages Number of Succeeded String of the failed numbers, use semicolon as the sperator
Return 2 Return message	Failed Messages

5 JavaScript API

By calling a JavaScript file on asterCC server, you can use javascript function to send a request to CTI. Via JavaScript API, you can control agent status and calls.

JS File URL: `http://$(ipaddress):$(port)/asterccinterface/astcc_CJI.js`

Introduction:

`$(ipaddress)` — IP address of the astercc server

`$(port)` — the port of the HTTP service (it is 80 by default)

System provides a example on [http://\\$\(ipaddress\):\\$\(port\)/asterccinterface/test_CJI.html](http://$(ipaddress):$(port)/asterccinterface/test_CJI.html)

5.1 Agent Login

Function:

loginCJI(orgidentity,usertype,user,pwdtype,password,callbackFuc)

Description:

Send a login request to asterCC;

Parameters:

Params	Type	Description
orgidentity	String	The identity string of the team
usertype	String	agent / account
user	String	Agent No. or Username
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
callbackFuc	String	Call back function

Return Format:

{code:code,message:message,param:param}

Type of the Return Value: JSON

Return Values:

返回值	Description
{code:1,message:msg,param:param}	{code: Succeeded ,message: Messages ,param: Parmas }
{code:2,message:msg }	{code: Failed ,message: Messages }
{code:3,message:msg }	{code: Failed ,message: Already Login }

param is a string, it will end with ||| for each agent status

Example:

Param='agent_group_id=1&groupname=Japanese&status=idel&groupadmin=no
&agenttype=static&agentline=online&errorcall=no&agent_id=10|||';

agent_group_id: agent group id

groupname: agent group name

status: agent status in the agent group, idel, ringing, busy, pause

groupadmin: If the agent is admin in the group: yes, no

agenttype: static, dynamic

agentline: online, offline

errorcall: If the agent has privilege to delete current call records: yes, no

agent_id: agent id in database

5.2 Agent Logout

Function:

logoutCJI(orgidentity,usertype,user,pwdtype,password,callbackFuc)

Description:

Send a logout request to asterCC;

Parameters:

Params	Type	Description
orgidentity	String	The identity string of the team
usertype	String	agent / account
user	String	Agent No. or Username
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
callbackFuc	String	callback

Return Format:

{code:code,message:message}

Type of the Return Value:JSON

Return Values:

Value	Description
{code:1,message:msg }	{code:Succeed,message:Messages}
{code:2,message:msg }	{code:Failed,message:Messages}

5.3 Agent group (queue) login/logout

Function:

queueActionCJI(type, usertype, user, orgidentity, list, pwdtype, password, pushevent, callbackFuc)

Description:

Agent login for one or several agents group

Parameters:

Params	Type	Description
type	String	1(login) / 2/logout
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team
list	String	Put a string combine with agentgroup id, end with a comma, if you don't put the list, it will login/logout from all agent group the agent belongs to
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
pushevent	String	If need send a agent event, yes / no(default)
callbackFuc	String	callback function

A valid list format example: agentgroupid1,agentgroupid2,.....agentgroupidn,

Return Format:

{code:code,message:message}

Type of the Return Value:JSON

Return Values:

Value	Description
{code:1,message:message}	{code:Succeed,message:Messages}
{code:2,message:message}	{code:Failed,message:Messages}

5.4 Agent group (queue) Pause/Unpause

Function:

queuePauseCJI(type, usertype, user, orgidentity, pwdtype, password, pause_reason, pushevent, callbackFuc)

Description:

Pause/Unpause the agent (only when he is login)

Parameters:

Params	Type	Description
type	String	1(Pause)/2(Unpause)
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
pause_reason	String	Put a pause reason when it's for pause Use the following string, if you need statistics training / meeting / leave / lunch / rest / other
pushevent	String	If need send a agent event, yes / no(default)
callbackFuc	String	callback function

Return Format:

{code:code,message:message}

Type of the Return Value:JSON

Return Values:

Value	Description
{code:1,message:message}	{code:Succeed,message:Messages}
{code:2,message:message}	{code:Failed,message:Messages}

5.5 ACW (After Call Work) Mode Config

Function:

acwActionCJI(type, usertype, user, orgidentity, pwdtype, password, agent_group_id, pushevent, callbackFuc)

Description:

Change the ACW mode of the agent in agent group

Parameters:

Params	Type	Description
type	String	1(Ringing ACW) / 2(Answered ACW) / 3(Disable ACW)
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
agent_group_id	String	agent group id
pushevent	String	If need send a agent event, yes / no(default)
callbackFuc	String	callback function

ACW Mode, when agent is in ACW mode, agent would not receive any calls from queue.

- n** Ringing ACW: As long as the agent phone rings, no matter if agent answer, when the call hangup, agent will be in ACW mode
- n** Answered ACW: Only the call is answered, agent will be in ACW mode when call hangup.
- n** Disable ACW: agent would not go to ACW mode automaticly

Return Format:

{code:code,message:message}

Type of the Return Value:JSON

Return Values:

Value	Description
{code:1,message:message}	{code:Succeed,message:Messages}
{code:2,message:message}	{code:Failed,message:Messages}

5.6 Exit ACW Mode

Function:

acwOffCJI(sessionid, usertype, user, orgidentity, pwdtype, password, pushevent, callbackFuc)

Description:

Exit the ACW Mode if agent is in

Parameters:

Params	Type	Description
sessionid	String	Session id of the current call (you can get this from asterCC events)
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
pushevent	String	If need send a agent event, yes / no(default)
callbackFuc	String	callback

Return Format:

{code:code,message:message}

Type of the Return Value:JSON

Return Values:

Value	Description
{code:1,message:message}	{code:Succeed,message:Messages}
{code:2,message:message}	{code:Failed,message:Messages}

5.7 Workway Mode Config

Function:

workwayActionCJI(status, usertype, user, orgidentity, pwdtype, password, agent_group_id, pushevent, callbackFuc)

Description:

Change the agent work mode in agent group

Parameters:

Params	Type	Description
status	String	dialin (Dial in only) / dialout (Dial out only) / all
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
agent_group_id	String	ids of the agent group
pushevent	String	If need send a agent event, yes / no(default)
callbackFuc	String	callback function

Return Format:

{code:code,message:message}

Type of the Return Value:JSON

Return Values:

Value	Description
{code:1,message:message}	{code:Succeed,message:Messages}
{code:2,message:message}	{code:Failed,message:Messages}

5.8 Originate Call

Function:

makeCallCJI(targetdn, targettype, agentgroupid, usertype, user, orgidentity, pwdtype, password, modeltype, model_id, userdata, callbackFuc)

Description:

Originate a call

Parameters:

Params	Type	Description
targetdn	String	Agent No. (if it's a internal call, and they must belong to same team) or put phone number if it's an external call
targettype	String	exter (external call) / inner (internal call)
agentgroupid	String	id of the agent group
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
modeltype	String	BusinessApp / Campaign / Virtualcustomer / Customer-service
model_id	String	id of the model which you want to originate the call for
userdata	String	user data, only valid when modeltype= Campaign, say you put "customer_id=109&pwd=asdfg" here, when you receive the call events, you will get userdata='customer_id=109&pwd=asdfg'
callbackFuc	String	callback function

Return Format:

{code:code,message:message}

Return Type:JSON

Return Values:

Value	Description
{code:1,message:message}	{code:Succeed,message:Messages}
{code:2,message:message}	{code:Failed,message:Messages}

5.9 Call Consult

Function:

consultCJI(targetdn, agentgroupid, consulttype, agentuniqueid, calleruniqueid, pwdtype, password, usertype, user, orgidentity, callbackFuc)

Description:

When agent in a call or in a conference, make a call to a new party, when it's answered, agent will talk to new party, the old party will be held(MOH) or stay in conference

Parameters:

Params	Type	Description
targetdn	String	Agent No. (internal consult) or phone number (external consult)
agentgroupid	String	agent group id when it's a internal consult, put 0 if it's a external consult
consulttype	String	internal / external
agentuniqueid	String	agent uniqueid of the agent
calleruniqueid	String	uniqueid of the caller
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team
callbackFuc	String	callback function

Return Format:

{code:code,message:message}

Type of the Return Value:JSON

Return Values:

Value	Description
{code:1,message:message}	{code:Succeed,message:Messages}
{code:2,message:message}	{code:Failed,message:Messages}

5.10 Call Transfer

Function:

transferCJI(sessionid, nwaycalluniqueid, calleruniqueid, agentuniqueid, pwdtype, password, usertype, user, orgidentity, callbackFuc)

Description:

When call consult success, use call transfer will connect the consult and the call, and the agent will be hangup from the call.

Parameters:

Params	Type	Description
sessionid	String	sessionid of the current call
nwaycalluniqueid	String	uniqueid of the consult party
calleruniqueid	String	uniqueid of the caller
agentuniqueid	String	uniqueid of the agent
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team
callbackFuc	String	callback function

Return Format:

{code:code,message:message}

Type of the Return Value:JSON

Value	Description
{code:1,message:message}	{code:Succeed,message:Messages}
{code:2,message:message}	{code:Failed,message:Messages}

5.11 End a Call Consult

Function:

callReturnCJI(nwaycalluniqueid, calleruniqueid, agentuniqueid, pwdtype, password, usertype, user, orgidentity, callbackFuc)

Description:

When call consult success, agent hangup the consult party and go back to the call.

Parameters:

Params	Type	Description
nwaycalluniqueid	String	uniqueid of the consult party
calleruniqueid	String	uniqueid of the caller
agentuniqueid	String	uniqueid of the agent
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team
callbackFuc	String	callback function

Return Format:

{code:code,message:message}

Type of the Return Value:JSON

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

5.12 Conference

Function:

conferenceCJI(nwaycalluniqueid, calleruniqueid, agentuniqueid, consultcount, pwdtype, password, usertype, user, orgidentity, callbackFuc)

Description:

When call consult success, connect the consult party to the call

Parameters:

Params	Type	Description
nwaycalluniqueid	String	uniqueid of the consult party
calleruniqueid	String	uniqueid of the caller
agentuniqueid	String	uniqueid of the agent
consultcount	String	how many parties in the conference besides caller and agent
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team
callbackFuc	String	callback function

Return Format:

{code:code,message:message}

Type of the Return Value:JSON

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

5.13 Hangup

Function:

hangupCJI(uniqueid, pwdtype, password, usertype, user, orgidentity, callbackFuc)

Description:

Hangup one call party or a list of parties

Parameters:

Params	Type	Description
uniqueid	String	the uniqueid of the calls you want to hangup, end with a comma
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team
callbackFuc	String	callback function

uniqueid example:

Hangup one party: uniqueid,

Hnagup: uniqueid1, uniqueid2, ...uniqueidN,

Return Format:

{code:code,message:message}

Type of the Return Value:JSON

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

5.14 Barge-in

Function:

intrudeCJI(sessionid, phonenumber, pwdtype, password, usertype, user, orgidentity, callbackFuc)

Description:

Agent Group admin could barge-in to a call

Parameters:

Params	Type	Description
sessionid	String	sessionid of the call
phonenumber	String	the phone number you want to use for barge-in, you can use a extension in system or any phone number
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team
callbackFuc	String	callback function

Return Format:

{code:code,message:message}

Type of the Return Value:JSON

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

5.15 Spy

Function:

silentMonitorCJI(sessionid, phonenumber, pwdtype, password, usertype, user, orgidentity, callbackFuc)

Description:

Agent Group admin could spy to a call

Parameters:

Params	Type	Description
sessionid	String	sessionid of the call
phonenumber	String	the phone number you want to use for spy, you can use a extension in system or any phone number
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team
callbackFuc	String	callback function

Return Format:

{code:code,message:message}

Type of the Return Value:JSON

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

5.16 Agent Status

Function:

agentStatus(orgidentity, usertype, user, pwdtype, password)

Description:

Get agent status

Parameters:

Params	Type	Description
orgidentity	String	The identity string of the team
usertype	String	agent / account
user	String	Agent No. or Username
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
callbackFuc	String	callback function

Return Format:

|Return|code|Return|message|Return|status

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message Return status	Succeeded Messages Status String (when agent belongs to several agent group, it will use comma as separator)
Return 2 Return message	Failed Messages

format of statusstring:agentgroupid1-status1,agentgroupid2-status2,.....

agentgroupidn-statusn,

5.17 Whisper

Function:

whisperCJI (sessionid, phonenumber, pwdtype, password, usertype, user, orgidentity, callbackFuc)

Description:

Agent Group admin could talk to agent in a call, while all the other parties in the call can not hear

Parameters:

Params	Type	Description
sessionid	String	sessionid of the call
phonenumber	String	the phone number you want to use for whisper, you can use a extension in system or any phone number
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team
callbackFuc	String	callback function

Return Format:

{code:code,message:message}

Type of the Return Value:JSON

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

5.18 Agents Group Stauts

Function:

agentgroupStatusCJI (orgidentity, usertype, user, pwdtype, password, agent_group_id, agenttype, callbackFuc)

Description:

Agent Group admin could get the status of the agents in his group

Parameters:

Params	Type	Description
orgidentity	String	The identity string of the team
usertype	String	agent / account
user	String	Agent No. or Username
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
agent_group_id	String	id of the agent group
agenttype	String	login(only check agents who is login) / all (all agents in the agent group)
callbackFuc	String	callbackfunction

Return Format:

{code:code,message:message,status:status}

Type of the Return Value:JSON

Return Values:

Return	Description
Return 1 Return message Return status	Succeeded Messages Status String
Return 2 Return message	Failed Messages

Status String contains the status of one or several agents, each status will end with ||

for one agent: agent no,agent id,status||

for several agents:

agentno1-agentid1,status1||agentno2-agentid2,status2||.....agentnoN-agentidN,statusN||

Status: idle, ringing, busy, in use(agent have hangup, but the conversation is not, like caller is talking to a consult, but agent is free to make call), pause,logout

agentno1-agentid1,status1||agentno2-agentid2,status2||.....agentnoN-agentidN,statusN||

5.19 Call Release

Function:

forcedRelease(sessionid,phonenumber,usertype,user,pwdtype,password,orgidentity)

Description:

Agent group admin will hangup the agent and talk to caller

Parameters:

Params	Type	Description
sessionid	String	sessionid of the call
phonenumber	String	the phone number you want to use for call release, you can use a extension in system or any phone number
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
usertype	String	agent / account
user	String	Agent No. or Username
orgidentity	String	The identity string of the team
callbackFuc	String	callback unction

Return Format:

|Return|code|Return|message|Return|status

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

5.20 Hold a Call

Function:

holdCJI (agentuniqueid, calleruniqueid, orgidentity, usertype, user, pwdtype, password, callbackFuc)

Description:

When it's two parties call, agent could put caller on hold, then caller will listen music

Parameters:

Params	Type	Description
agentuniqueid	String	call uniqueid of the agent
calleruniqueid	String	call uniqueid of the caller
orgidentity	String	The identity string of the team
usertype	String	agent / account
user	String	Agent No. or Username
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
callbackFuc	String	callback function

Return Format:

{code:code,message:message}

Type of the Return Value:JSON

Return Values:

Value	Description
{code:1,message:msg,status:status}	{code:成功,message:信息描述}
{code:2,message:msg}	{code:失败,message:信息描述}

5.21 Unhold a Call

Function:

resume(agentuniqueid,calleruniqueid,orgidentity,usertype,user,pwdtype,password)

Description:

Get back the onhold call

Parameters:

Params	Type	Description
agentuniqueid	String	call uniqueid of the agent
calleruniqueid	String	call uniqueid of the caller
orgidentity	String	The identity string of the team
usertype	String	agent / account
user	String	Agent No. or Username
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
callbackFuc	String	callback function

Return Format:

|Return|code|Return|message|Return|status

Type of the Return Value: String

Return Values:

Value	Description
Return 1 Return message	Succeeded Messages
Return 2 Return message	Failed Messages

5.22 Predictive Dialer

Function:

dialerlist (orgidentity, usertype, user, pwdtype, password, campaignid, phonenum, priority, dialtime)

Description:

Add phone numbers to the predictive dial list of a campaign

Parameters:

Params	Type	Description
orgidentity	String	The identity string of the team
usertype	String	agent / account
user	String	Agent No. or Username, the user must be group admin of the campaign
pwdtype	String	Use plaintext or md5 password for agent or account
password	String	password
campaignid	String	id of the campaign
phonenum	String	phone numbers, use semicolon as the separator when you have more than one. You can put 100 phone numbers at most, we suggest less than 50
priority	String	priority of the phone number in predictive dialer
dialtime	String	the dial time, if you give a empty string, it would be 0000-00-00 00:00:00
callbackFunc	String	callback function

Return Format:

|Return|code|Return|message

Type of the Return Value:String

Return Values:

Value	Description
Return 1 Return message Return succ Return err	Succeeded Messages Number of Succeeded String of the failed numbers, use semicolon as the separator
Return 2 Return message	Failed Messages

6 Receive events via javascript

System provides a example code, you can find from

[http://\\$\(ipaddress\):\\$\(port\)/asterccinterface/astercc_nginx_http_push.html](http://$(ipaddress):$(port)/asterccinterface/astercc_nginx_http_push.html)

To receive events of a team in javascript, you need go to Team -> Advanced -> API domain, put 127.0.0.1

Restart CTI as admin, or execute linux command /etc/init.d/asterccd restart

Step 1: Call the js file from asterCC server in your web page

```
<script src="
```

```
http://$(ipaddress):$(port)/asterccinterface/astercc_nginx_http_push.js "></script>
```

By include this js file, you web application will get the ability to receive astercc events

Step 2: Create the javascript function sonAccept(message)

```
function sonAccept(message) {
```

```
}
```

Description:

This function is used to process events sent from the CTI in realtime.

sonAccept: function name

message: events, string

a sample message string:

```
"calleridnum=041139735857&didnumber=8008008&activenum=041139735855&source=AGENT&event=ringing&uniqueid=78969&sessionid=efeb7b374a3408b0d0954f30a1504d83&eventTime=2010-01-01 08:21:58&AgentTeamId=20&modeltype=Campaign&model_id=10&AgentGroupId=100&calltype=dialout&channel=Local/015967121144@hosted-dialout-408a;1&from=dialout&";
```

The following code could be used to parse the string to array:

```
var aryMessage = message.split('&');
```

```
var aryEvent = new Array();
```

```
for(i=0;i<aryMessage.length;i++) {
```

```
var tmp = aryMessage[i].split('=');
```

```
eval("aryEvent['"+tmp[0]+"'] = '"+tmp[1]+"");  
}
```

Then you get the js array anyEvent, easier to process.

Step 3: Process events

Now you get the events, you can do what you need, like popup a window, change statusbar etc.

Here is a list of events for some typical call flow, events will be sent ordinally.

6.1 Dial out:

6.1.1 Agent ring:

message=

```
"calleridnum=013888888888&didnumber=&activenum=041139735857&source=AGENT  
&event=ringing&uniqueid=19866&sessionid=efeb7b374a3408b0d0954f30a1504d83&eventTime=2010-01-01  
08:22:01&AgentTeamId=20&modeltype=Campaign&model_id=10&AgentGroupId=100&calltype=dialout&channel=Local/041139735857@hosted-dialout-408a;1&from=dialout&";
```

6.1.2 Agent answer:

message=

```
"calleridnum=013888888888&didnumber=&activenum=041139735857&source=AGENT  
&event=answer&uniqueid=19866&sessionid=efeb7b374a3408b0d0954f30a1504d83&eventTime=2010-01-01  
08:22:01&AgentTeamId=20&modeltype=Campaign&model_id=10&AgentGroupId=100&calltype=dialout&channel= &from=dialout&";
```

6.1.3 Customer ring:

message=

```
"calleridnum=013888888888&didnumber=&activenum=041139735857&source=CALLEE  
&event=ringing&uniqueid=19867&sessionid=efeb7b374a3408b0d0954f30a1504d83&eventTime=2010-01-01  
08:22:01&AgentTeamId=20&modeltype=Campaign&model_id=10&AgentGroupId=100&c
```

alltype=dialout&channel=Local/013888888888@hosted-dialout-7636;1&from=dialout&”
;

6.1.4 Customer answer:

message=

“calleridnum=041139735857&didnumber=&activenum=041139735857&source= CALLEE
&event=answer&uniqueid=19867&sessionid=efeb7b374a3408b0d0954f30a1504d83&eve
ntTime=2010-01-01
08:22:09&AgentTeamId=20&modeltype=Campaign&model_id=10&AgentGroupId=100&c
alltype=dialout&channel=&from=dialout&”;

6.1.5 Customer Hangup:

message=

“calleridnum=013888888888&didnumber=&activenum=041139735857&source= CALLEE
&event=hangup&uniqueid=19867&sessionid=efeb7b374a3408b0d0954f30a1504d83&ev
entTime=2010-01-01
08:22:47&AgentTeamId=20&modeltype=Campaign&model_id=10&AgentGroupId=100&c
alltype=dialout&channel=Local/013888888888@hosted-dialout-7636;1&from=dialout&”
;

6.1.6 Agent hangup:

message=

“calleridnum=013888888888&didnumber=&activenum=041139735857&source= AGENT
&event=hangup&uniqueid=19866&sessionid=efeb7b374a3408b0d0954f30a1504d83&ev
entTime=2010-01-01
08:22:47&AgentTeamId=20&modeltype=Campaign&model_id=10&AgentGroupId=100&c
alltype=dialout&channel=Local/041139735857@hosted-dialout-408a;1&from=dialout&”;

6.1.7 Conversation hangup:

message=

“calleridnum=013888888888&didnumber=&activenum=041139735857&source= CONVE
RSA-
TION&event=hangup&uniqueid=0&sessionid=efeb7b374a3408b0d0954f30a1504d83&ev

entTime=2010-01-01

08:22:47&AgentTeamId=20&modeltype=Campaign&model_id=10&AgentGroupId=100&calltype=dialout&channel=&from=dialout&";

6.2 Dial in:

6.2.1 Customer dialin:

message= "CALL-

ER=incoming&modeltype=Campaign&uniqueid=1555&model_id=10&from=dialin&sessionid=1f10eab01f5b1b36230b8e91aef34810&channel=Local/013888888888@hosted-dialin-701a;1";

6.2.2 Agent ring:

message=

"calleridnum=013888888888&didnumber=800888&activenum=041139735857&source=AGENT&event=ringing&uniqueid=1556&sessionid=1f10eab01f5b1b36230b8e91aef34810&eventTime=2010-01-01 11:11:07&AgentTeamId=3&modeltype=Campaign&model_id=10&AgentGroupId=100&from=dialin&channel=Local/041139735857@hosted-dialin-531a;1";

6.2.3 Agent answer:

message=

"calleridnum=013888888888&didnumber=800888&activenum=041139735857&source=AGENT&event=answer&uniqueid=1556&sessionid=1f10eab01f5b1b36230b8e91aef34810&eventTime=2010-01-01 11:11:10&AgentTeamId=3&modeltype=Campaign&model_id=10&AgentGroupId=100&from=dialin&channel="";

6.2.4 Customer hangup:

message=

"calleridnum=013888888888&didnumber=800888&activenum=041139735857&source=CALL-ER&event=hangup&uniqueid=1555&sessionid=1f10eab01f5b1b36230b8e91aef34810&ev

entTime=2010-01-01

11:12:54&AgentTeamId=3&modeltype=Campaign&model_id=10&AgentGroupId=100&from=dialin&channel=Local/0138888888888888@hosted-dialin-701a;1";

6.2.5 Agent hangup:

message=

"calleridnum=0138888888888&didnumber=800888&activenum=041139735857&source=AGENT&event=hangup&uniqueid=1556&sessionid=1f10eab01f5b1b36230b8e91aef34810&eventTime=2010-01-01

11:12:54&AgentTeamId=3&modeltype=Campaign&model_id=10&AgentGroupId=100&from=dialin&channel=Local/041139735857@hosted-dialin-531a;1";

6.2.6 Conversation hangup:

message=

"calleridnum=0138888888888&didnumber=&activenum=041139735857&source=CONVERSATION&event=hangup&uniqueid=0&sessionid=1f10eab01f5b1b36230b8e91aef34810&eventTime=2010-01-01

11:12:54&AgentTeamId=3&modeltype=Campaign&model_id=10&AgentGroupId=100&channel=&from=dialin";

Parameters:

source: source of the event, it could be AGENT, CALLER, CALLEE, CONVERSATION, CONSULT

***all source would be UPCASE**

event: ringing, answer, hangup, join, incoming, onhold, resume

***all event would be lowercase**

calleridnum: the caller id number of the caller or callee

activenum: the caller id number of the source party

didnumber: DID number of the call

eventTime: time when the event raise

sessionid: all events in same conversation will have same sessionid,and it would be unique in the whole system

from: "dialin" when it's inbound call, "dialout" when it's outbound

eventid: sequence number of the event, when the eventid of a event is smaller than the event id you record, then it means it was an old event

uniqueid: unique id of the source party

channel: channel string of the source party

AgentTeamId: team id of the event

AgentGroupId: agent group id of the event

modeltype: which model the event belongs to, now we have model Campaign, Virtualcustomer, Customerservice, BusinessApp

model_id: id of the model

7 Receive events via http post

In asterCC you can config a URL, asterCC CTI will post the event to the URL, http post is mainly used to system control when JS API is used for agent control.

Example:

Step 1: Create a php page to receive events and generate CDR

We need read all events for a call, then we could find out how to generate a CDR

here's a outbound call example for php+mysql, asterCC will post the events to the code:

Click-2-call scenario analysis:

1. Agent send a request to CTI, I need to call number XXXX
 2. CTI will first call agent, agent phone ring
 3. Agent answer
- * only after agnet answer, CTI will call the caller
4. CTI call customer, customer ring (in php code, it will create a call record, ringing)
 5. Customer answer (update the record to answered)
 6. Either agent or customer hangup, conversation hangup (update record)

In aboce scenario, we will know

Agent No.: \$agentno

Agent Phone Num.: \$agentphone

Team Identity: \$orgidentity

We will know these information when agent login

The conversation will have severail call events, and they will have same sessionid , and we can know call status via source and event

```
function cdr() {  
    if($data ['calltype'] == 'dialout'){  
        //if it's dial out  
        if($data['source'] == 'CALLEE' && $data['event'] == 'ringing'){  
            //agent originate a call, agent answer, then customer phone will ring, we create  
            the record here  
            $src = (string)$agentphone; //CALLER number, agent
```

```
$dst = (string)$data['activenum'];//CALLEE number
$starttime = $data['eventTime'];//call start time
$disposition = 'NOANSWER';//customer ring, we use noanswer
$target = 'DialOut';// outbound call
$diallogid = $data['sessionid']; //unique session id
$agentno;// save Agent No. if you need
$orgidentity;//save team identify string if you need
If(! (SELECT * FROM YOURCDR WHERE diallogid=$diallogid limit 1;)) {
    //check if we already created the record, or you can set unique filed for di-
allogid
    INSERT INTO YOURCDR SET YOURFIELDS;//create the record
}
}else if($data['source'] == 'CALLEE' && $data['event'] == 'answer'){
    //customer answer
    $disposition = 'ANSWER';//update status
    $answertime = $data['eventTime'];//set answer time
    UPDATE YOURCDR SET YOURFIELDS WHERE diallogid=$diallogid limit 1;
}else if($data['source'] == 'CALLEE' && $data['event'] == 'hangup'){
    $endtime = $data['eventTime'];//call end time
    UPDATE YOURCDR SET YOURFIELDS WHERE diallogid=$diallogid limit 1;
}
}
}
```

Step 2: Config the address so asterCC knows where to post the event

Go to Team -> Advanced -> Push event URL, put your php code url

Restart CTI as admin, or execute linux command /etc/init.d/asterccd restart

Step 3: Process events

When your program start receiving events, you can use exactly same function in Chap. 6 to process these events, such as CDR generation.

8 Appendix

8.1 Language Code

Code	Language
cn	Simplified Chinese
en	English
jp	Japanese
kr	Korean
Ru	Russian
fr	French
de	German
es	Spanish

8.2 Error Codes

Code	Description
BackMsg_01	Parameters are empty
BackMsg_02	Team doesn't exist, identity string is wrong
BackMsg_03	Agent No. doesn't exist or password is wrong
BackMsg_04	Succeed to login
BackMsg_05	Agent account is suspend
BackMsg_06	Agent account doesn't exist
BackMsg_07	Password is wrong
BackMsg_08	Succeed to logout
BackMsg_09	Agent No. doesn't exist [Agent No.]
BackMsg_10	Failed to switch agent group status
BackMsg_11	Agent group id doesn't exist [agent group id]
BackMsg_12	Agent doesn't exist
BackMsg_13	Variable <type> is invalid
BackMsg_14	Succeed to switch agent group status
BackMsg_15	Account doesn't exist
BackMsg_16	Failed to call, agent is not check-in 入
BackMsg_17	Failed to call, agent status is [status]
BackMsg_18	Failed to call, agent is limited to make outbound call
BackMsg_19	Succeed to call
BackMsg_20	Failed to call
BackMsg_21	Failed to consult
BackMsg_22	There is already a call request pending
BackMsg_23	Agent doesn't bind to any extension, can't call
BackMsg_24	Agent is not found, can't call
BackMsg_25	The agent you want to consult is not check-in
BackMsg_26	Failed to create consult record
BackMsg_27	Succeed to consult
BackMsg_28	Parameters are wrong
BackMsg_29	Failed to transfer
BackMsg_30	Agent group doesn't exist

BackMsg_31	Succeed to transfer
BackMsg_32	Succeed to end consult
BackMsg_33	Failed to end consult
BackMsg_34	Succeed to start the conference
BackMsg_35	Failed to start the conference
BackMsg_36	Failed to hangup
BackMsg_37	Succeed to hangup
BackMsg_38	No account for the agent
BackMsg_39	Succeed to barge-in
BackMsg_40	Failed to barge-in
BackMsg_41	Succeed to spy
BackMsg_42	Failed to spy
BackMsg_43	Succeed to get data
BackMsg_44	Failed to get data, no records match
BackMsg_45	You can't release the call
BackMsg_46	Succeed to release
BackMsg_47	Failed to release
BackMsg_48	Succeed to whisper
BackMsg_49	Failed to whisper
BackMsg_50	Agent group doesn't exist in the team
BackMsg_51	Failed to hold the call
BackMsg_52	Succeed to hold the call
BackMsg_53	Failed to unhold the call
BackMsg_54	Succeed to unhold the call
BackMsg_55	Agent No. is wrong or doesn't exist
BackMsg_56	Can't consult, agent status is [status]
BackMsg_57	No agent group to check-in
BackMsg_58	No check-in in any group
BackMsg_59	Succeed to swith the ACW mode
BackMsg_60	Agent is not check-in or doesn't exist in the group
BackMsg_61	Succeed exit ACW mode
BackMsg_62	Succeed switch the work way
BackMsg_63	It is usertype=account in the request, but the account doesn't exist or the password is wrong

BackMsg_64	Please check if the uniqueid is valid
BackMsg_65	Agent is not group admin
BackMsg_66	Agent is already login