



# API (Phone Mashup)

Developer Guide



## Contents

- Introduction to the Documentation .....3**
  - About the Documentation .....3
  - Ifbyphone on the Web.....3
- Advanced Techniques for Developers.....4**
  - Locating IDs.....4
- Smart Click-to-Call API.....6**
  - Trigger a Smart Click-to-Call Created with the Wizard.....6
  - Trigger an ifbyphone Object by Passing Parameters via URL.....7
  - Trigger a Call between Two Unregistered Numbers.....7
  - Typical Formats for Smart Click-to-Call URLs.....7
  - Language Specific Examples .....8
  - Parameters for Smart Click-to-Call Configurations .....9
  - SurVo Variable Substitution.....12
  - NetGet Overview.....14
    - Configuring the NetGet.....14
  - The NetGet Page.....15
    - NetGet Actions.....16
    - Submit Type .....17
    - Domain.....17
    - Page .....17
    - Static Parameters.....17
    - Dynamic Parameters.....17
  - Example: NetGet.....17
  - Trigger a Reminder Call.....20
  - Set a Schedule Mode.....20
- Administrative API.....22**
  - Voice Mail .....23
  - Report .....24
  - SurVo .....24
  - Verify-Me-Now.....25
  - Find Me.....25
  - Voice Broadcast.....29
- Ifbyphone Glossary .....31**



## Introduction to the Documentation

*This guide defines a set of calling conventions for implementation of the services invoked through the Smart Click-to-Call and SurVo building blocks.*

The *API (Phone Mashup) Developers Guide* has been created to let clients and developers interact programmatically with Ifbyphone's building blocks.

Advanced techniques allow you to perform simple functions like triggering outbound calls, as well as build more complex voice applications, such as chaining together a series of SurVos, which can be controlled by your remote server using almost any programming language.

### About the Documentation

The *API (Phone Mashup) Developers Guide* includes the following documentation:

- Descriptions of the Smart Click-to-Call and the Administrative APIs
- Code samples
- Screenshots and helpful tips for productivity

ICON KEY	
	Note
	Warning

Throughout the document, icons are used to indicate important reminders, including tips and common mistakes. Screenshots are provided to guide users through implementing procedures outlined in this manual.

### Ifbyphone on the Web

Ifbyphone's Web site ([www.ifbyphone.com](http://www.ifbyphone.com)) contains a wealth of information about Ifbyphone business services. Customer service contact information, as well as additional user guides and tutorials, are also easily accessible from the Web site.

# 2

## Advanced Techniques for Developers

The techniques contained in this section allow developers to create database-generated links and URLs, initiate server-triggered events and administer sophisticated SurVo functionality.

### Locating IDs

To obtain the ID for any of your Smart Click-to-Call objects (e.g. a Virtual Receptionist menu, Find Me list, SurVo, etc.), select the **Building Block IDs** menu option from the main page's top navigation bar under **Tools** (Figure 1). You will also find the public and API keys that are required for API calls.

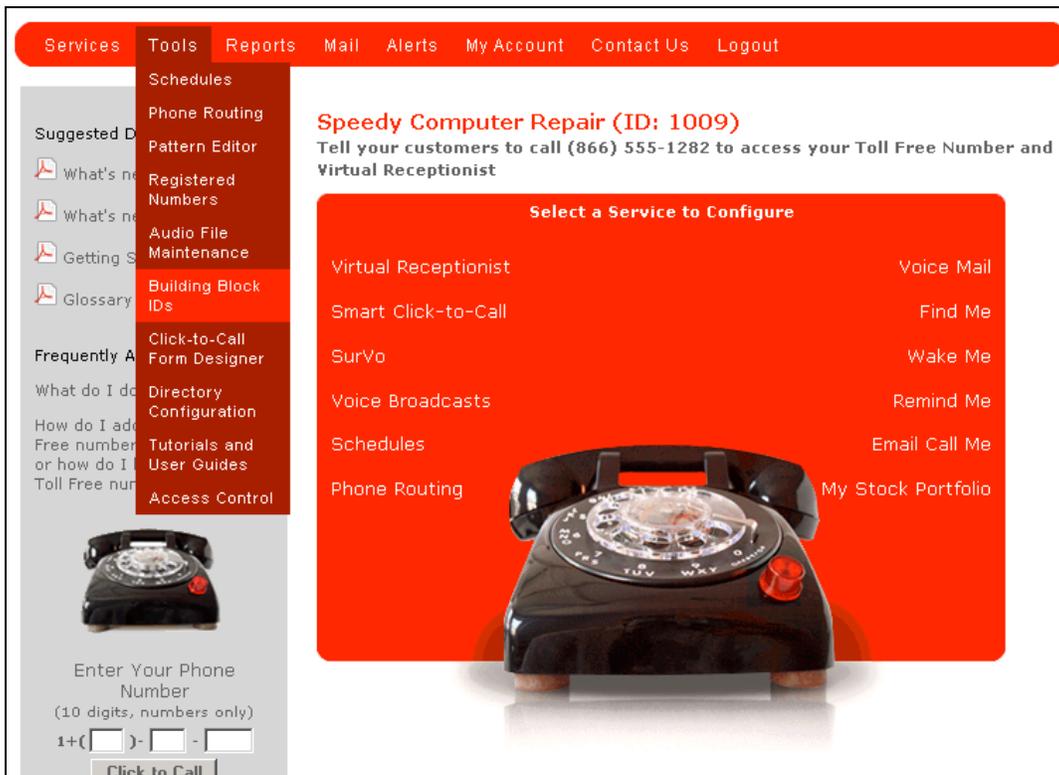


Figure 1: IDs for Smart Click-to-Call API Use

Your Building Block IDs will be listed by object (Figure 2).

**Building Block IDs for 'Greenberg Inc.'**

You can use the values listed in each Building Block's ID column for techniques described in the [Advanced Techniques for Developers](#) tutorial, as well as for utilization in importing [Phone Directory](#) items from CSV files, providing you with complete control over intelligent transfers to telephone numbers and supported Building Blocks of that service in one step.

**Your Public Key (use with click\_to links) =**  
296[REDACTED]361

**Your API Key (use with api calls) =** ccd1[REDACTED]90e

**Your Private Key =** 077[REDACTED]9e6

---

**Virtual Receptionist Menus (app: CTVR)**

ID	Menu Name
62	WELCOME
70	ADAM GREENBERG
17	VACATION
80	SALES GUY ONE
80	SALES GUY TWO
80	SALES GUY THREE
21	TRANSFER
22	NEW MEXICO

---

**Find Me Lists (app: CTF)**

ID	List Name
43	FIND TECHNICIAN
54	MOTHER

Figure 2: Click-to Object IDs

# 3

## Smart Click-to-Call API

*This guide defines a set of calling conventions for triggering outbound calls and other actions through the Smart Click-to-Call building block.*

### Uses for Smart Click-to-Call Advanced Programming Techniques

Using Smart Click-to-Call advanced programming techniques, you can trigger calls through Web sites on the ifbyphone system. Advanced techniques can be applied in three different ways:

- To trigger a Smart Click-to-Call that has been created with the Wizard
- To trigger an ifbyphone object by passing parameters via URL
- To trigger a call between two unregistered numbers

See the table in *Parameters for Smart Click-to-Call Configurations* for information on the different ifbyphone building blocks that can be triggered via the advanced techniques.

#### Trigger a Smart Click-to-Call Created with the Wizard

You can use the Web-based Wizard (see the *Smart Click-to-Call Advanced User's Guide* for more information) to create a Smart Click-to-Call object, as well as the code necessary for implementing a Smart Click-to-Call in an Email message, Email signature or Web page (Figure 3). Typically you will need just the `click_id` parameter and the public security key when activating a Smart Click-to-Call object created in this manner (see the table in *Parameters for Smart Click-to-Call Configurations* for `click_id` parameters).

User parameters or pass-through (`p_t`) data for Click-to-SurVo (voice survey) configurations can also be added to Wizard-generated code. See the section on SurVo Variable Substitution on page 10.

On the next page is a screen shot of the Smart Click-to-Call Wizard-Generated code.

**Image Pop-up** (HTML/Javascript) Code for putting a text or graphic link on your page that opens a new small window to capture the user's phone number.



```
'Clickto' , 'toolbar=no,location=no, menubar=no, scrollbars=no,
copyhistory=no,resizable=yes')">

</a>
```

---

**Input Box** (HTML/Javascript) Code for putting a input box and button on your page for capturing the user's phone number. When submitted, a small browser window with a page from the ifbyphone site will open with call status. With this approach, the user's current page remains visible in the browser.



```
<input type="text" id="phone" name="phone" maxlength="10"
size="10"><br>
<input type="submit" value="Click-to-Call"
onClick="window.open('http://www.ifbyphone.com/clickto_status.php?click
+ getElementById('phone').value, 'Clickto' ,
'width=200,height=200,toolbar=no,location=no, menubar=no,
```

---

**Just the URL** This is just the URL for getting to the page. If you are using a website builder it may only want the URL.

```
http://www.ifbyphone.com/clickto_getphone.php?click_id=109
```

**LINK**

Figure 3: Smart Click-to-Call Wizard-Generated Code

**Trigger an Ifbyphone Object by Passing Parameters via URL**

You can employ this method when the telephone numbers used in the Smart Click-to-Call phone call are database-driven. When activating a call between your business and a customer, one of the phone numbers in the call must be "registered" for use with ifbyphone. You can register the phone number via the ifbyphone Web site using the Registered Numbers utility (see the *Appendix* in the *Smart Click-to-XYZ Advanced User's Guide* for detailed instructions). You must then create a complete URL with all the necessary parameters (see the section on *Typical Formats for Smart Click-to-Call URLs* for examples).

**Trigger a Call between Two Unregistered Numbers**

In this situation, neither number is registered. Authorization credentials must be passed in the URL. It is strongly recommended that only a secure post method be used when employing this approach, as failure to do so could allow someone to acquire your account credentials. In addition to the normal parameters, a password parameter (pwd) must also be submitted.

**Typical Formats for Smart Click-to-Call URLs**

The following URL is an example of a typical format for a Smart Click-to-Call URL created via the Smart Click-to-Call wizard (where the ## characters will be substituted by the click\_id):

[https://secure.ifbyphone.com/click\\_to\\_xyz.php?click\\_id=##&phone\\_to\\_call=8475552222](https://secure.ifbyphone.com/click_to_xyz.php?click_id=##&phone_to_call=8475552222)

The following URL is an example of a typical format for a SurVo (voice survey) URL with user parameters and pass-through data (where the ## characters will be substituted by the click\_id. See the section on SurVo Variable Substitution on page 10 for information on using “URL Parameters):

[https://secure.ifbyphone.com/click\\_to\\_xyz.php?click\\_id=##&user\\_parameters=param1|value1||param2|value2||param3|value%20with%20multiple%20words&p\\_t=mydata%20to%20pass%20back&phone\\_to\\_call=847555222](https://secure.ifbyphone.com/click_to_xyz.php?click_id=##&user_parameters=param1|value1||param2|value2||param3|value%20with%20multiple%20words&p_t=mydata%20to%20pass%20back&phone_to_call=847555222)

The following URL is an example of a typical format for a Click-to-Call URL (where “id” will contain an ifbyphone-registered phone number and “phone\_to\_call” contains the phone number receiving the call):

[https://secure.ifbyphone.com/click\\_to\\_xyz.php?app=CTC&type=1&id=7734435562&acct=2834&phone\\_to\\_call=7735726922&pwd=9999](https://secure.ifbyphone.com/click_to_xyz.php?app=CTC&type=1&id=7734435562&acct=2834&phone_to_call=7735726922&pwd=9999)

To use an unregistered phone number, include a pwd parameter containing the password for the account.



#### **WARNING: Character Restrictions**

Pass-through data cannot contain any of the following characters: “&,” “=,” or “?” (i.e., ampersand, equal to, or question mark)

### *Language Specific Examples*

The following is an example of using .net to show a user a form and then use that input to trigger a call to their phone number:

```
<%@ Page Language="VB"%>
<%@ Import Namespace="System.Net" %>
<%@ Import Namespace="System.IO" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<script runat="server">
    Sub submit(ByVal sender As Object, ByVal e As EventArgs)
        Dim url As String

        url =
"http://secure.ifbyphone.com/click_to_xyz.php?key=d9vks0ewkgui8olqifjidob9e8d
uw0d08ro0w8i9r00e"
        url = url + "&phone_to_call=" + txtNPA.Text + txtNNX.Text +
txtLine.Text
        url = url + "&click_id=" + txtBuildingBlockID.Text
```

```

Dim req As HttpWebRequest = WebRequest.Create(url)

Try
    'Get the data as an HttpWebResponse object
    Dim resp As HttpWebResponse = req.GetResponse()

    Dim sr As New StreamReader(resp.GetResponseStream())
    Dim results As String = sr.ReadToEnd()
    sr.Close()

    lblCallStatus.Text = results
Catch wex As WebException
    Response.Write("<font color=red>Error<br />Status: " &
wex.Status & "Message: " & wex.Message & "</font>")
End Try
End Sub
</script>

<html>
<body>

```

This example shows the call to the server using a curl command in PHP:

```

$url="http://secure.ifbyphone.com/click_to_xyz.php?
phone_to_call
=
#####&click_id
==##&key=key=d9vks0ewkgui8olqifjidob9e8duw0d08ro0w8i9r00e";

//INITIATE CURL AND SET OPTIONS
    $session = curl_init();
    curl_setopt($session, CURLOPT_FOLLOWLOCATION , TRUE);
    curl_setopt($session, CURLOPT_RETURNTRANSFER , TRUE);
    curl_setopt($session, CURLOPT_HEADER, FALSE);

    curl_setopt($session, CURLOPT_URL, $url);

    $result = curl_exec($session);

    if (curl_errno($session)) {
        print("curl error:". curl_error($session));
    }

//TYPICALLY HERE YOU WOULD CHECK THE RESULT AND PUT HANDLING IF THE CALL DID
NOT GO THROUGH (BUSY, ETC.).
// A SUCCESSFUL CALL WOULD RETURN 'Call Connected'.

```

**Parameters for Smart Click-to-Call Configurations**

The following table lists each Click-to configuration type with its associated parameters and expected values.

Application	Parameter	Value/Notes
Click-to-Call	App	CTC
	type	1 = Call "phone_to_call" value first.

		2 = Call "registered number" first.
	id	A registered number or the main number of the account. Use the Registered Numbers Utility on the Tools menu to register a number for use in Click-to calls.
	phone_to_call	Telephone number to call containing digits only with no hyphens, spaces, or leading 1 (e.g. 8475551212).
	pwd	Required if the ID is not registered with ifbyphone.
	acct	The ifbyphone account ID.

<b>Click-to-Find Me</b>	app	CTF
	acct	The ifbyphone account ID.
	phone_to_call	Telephone number to call containing digits only with no hyphens, spaces, or leading 1 (e.g. 8475551212).
	findme_id	The ID of the selected Find Me list.

<b>Click-to-Find Me List</b> (This allows the developer to create a specific outbound message.)	app	CTFL
	list (required)	bar delimited list of phone numbers (e.g. 8475552323 8475554332)
	screen_prompt	Message that you want played, asking for the phone number. Should be encoded (e.g. tell%20us%20why%20you%20are%20calling)
	phone_to_call	Telephone number to call containing digits only with no hyphens, spaces, or leading 1 (e.g. 8475551212).
	nextaction	The "post action" that executes if Find Me fails to connect the caller. Currently accepted values are: 1 – Tell user no one is available and disconnect 2 – Transfer to Voice Mail 3 – Automatically accept call on last number tried 7 – Automatically accept call without intro
	nextactionitem	Currently only required for a "nextaction" value of 2. Must correspond to an Ifbyphone Voice Mail box found in the "Building Block ID" page that is owned by the specified Ifbyphone account ID.

(See next page for additional parameters)

<b>Click-to-Virtual Receptionist</b>	app	CTVR
	acct	The ifbyphone account ID.
	menu_id	The ID of the selected Virtual Receptionist menu.
	phone_to_call	Telephone number to call containing digits only with no hyphens, spaces, or leading 1 (e.g. 8475551212).

<b>Click-to-Voice Mail</b>	app	CTVM
	acct	The ifbyphone account ID.
	vmail_box_id	The ID for the voice mailbox.
	phone_to_call	Telephone number to call containing digits only with no hyphens, spaces, or leading 1 (e.g. 8475551212).

<b>Click-to-SurVo</b>	app	CTS
	acct	The Ifbyphone account ID.
	survo_id ("survo" is also accepted)	The ID of the selected SurVo.
	user_parameters	Optional list of parameters, in the form key value  key value. Values should be encoded. This entry is only used when scheduling a single "phone", not a " " delimited list or uploaded/fetched CSV file.
	p_t	Optional pass-through data (not used by the SurVo, but can be returned via a NetGet).
	phone_to_call	Telephone number to call containing digits only with no hyphens, spaces, or leading 1 (e.g. 8475551212) – not valid if "scheduleonly" flag is set.
	scheduleonly	Flag that tells the system that no calls are being placed immediately, but are performing Voice Broadcast API scheduling.

(See next page for additional parameters)

<p><b>Click-to-SurVo</b> Schedule only parameters</p>	phone	The phone number(s) scheduled to receive the SurVo Broadcast call. Not validated in the absence of the “scheduleonly” flag. May be a single phone (10 digit) phone number, a “[ ]” delimited list of (10 digit) phone numbers, or a fully qualified URL to a web-accessible and valid CSV file (must include at least a “number” column.)
	sdate	The official Voice Broadcast starting date/time in the format YYYY-MM-DD HH:MM (24 hour format).
	edate	The official Voice Broadcast ending date/time in the format YYYY-MM-DD HH:MM (24 hour format). Must be chronologically later than “sdate”. Note that this parameter is ignored if the ‘type’ parameter is set to “2” (As fast as possible).
	tz	The time zone all specified date/time values are considered to be in. (Valid values are “Eastern”, “Central”, “Mountain”, “Pacific”, “Alaska”, “Hawaii”; default is “Eastern”)
	dstime	The “Daily Call Range” starting time when Voice Broadcast calls can be made (Valid values meet the HH:MM 24 hour format)
	detime	The “Daily Call Range” ending time when Voice Broadcast calls can be made (Valid values meet the HH:MM 24 hour format and come later in the day than the value for “dstime”)
	type	The scheduling algorithm to use when scheduling calls for this Voice Broadcast. (Valid values are 1 for “Spread calls evenly” or 2 for “As fast as possible”)
	attempts	Maximum number of retry attempts for each phone number. (Valid values are 1,2,3; default is 1 – meaning no retries)
	retry	Number of minutes between retry attempts on a phone number. (Valid values are 5,10,15,30,60,90,120; default is 5)
	simul	Number of simultaneous calls to be placed every freq minutes. (Valid values are 1,2,3; default is 1)
	cid	Caller ID value that shows up to recipients of the Voice Broadcast. (Valid values are any phone number that matches one of the account’s toll free numbers, or is listed in the accounts Registered Phone number list; default is the account’s primary toll free number)
	desc	Identifier for the Voice Broadcast used for GUI management. (Valid values contain alphanumeric characters and spaces; default is “My SurVo Broadcast”)

**SurVo Variable Substitution**

You can dynamically insert information as the SurVo is presented to the respondent (see *Advanced SurVo User’s Guide* for more information).

**1. Define Variables in the SurVo**

Define your own variables or use built-in variables in the SurVo, including answers to previous questions. To define a variable, insert it into the question prompt or answer text, surrounded by curly brackets { }. For example, if you want to greet a customer with their first name, enter a prompt with:

*Hello {first\_name}!*

To insert the answer text from a previous question, use the form {A#} where # is the question number. For example, to reference an answer from Question 3, enter:

*“Why did you answer {A3}?”*

Other built-in variables include:

{callerid} – The phone number of the caller (if available), or, if outbound, the number called.

{unique\_id} – A unique reference ID assigned by the ifbyphone system.

## 2. Pass the Populated Data

Pass the parameters to the SurVo via a Smart Click-to-Call Wizard or in response to a SurVo NetGet. You can also pass the parameters in when doing an outbound SurVo.



### Note: Using Variables and Customized Audio

You can still use variable substitution, even if you choose to record your own audio prompts with your voice in place of using all text-to-speech. A single computerized message that reads “Hello {first name}, this is a call to remind you of your appointment on {date}” can be created using personalized voice messaging. Create four Prompt Only SurVo question types:

Question 1: Record an audio file that says “Hello.”

Question 2: In the Enter Question Text field, enter in the variable {first name}.

Question 3: Record an audio file that says “this is a call to remind you of your appointment.”

Question 4: In the Enter Question Text field, enter in the variable {date}.

### NetGet Overview

In addition to storing the results of completed SurVo forms in the ifbyphone database and sending via Email, the results of the SurVo can activate a NetGet. The NetGet transmits the SurVo results to the designated server. The action configured in the NetGet will determine what happens next. Possible outcomes include reading server-returned HTML or RSS formatted information to the respondent or performing an action listed in the returned XML data structure. To access the SurVo NetGet screen (Figure 4), please follow the steps outlined in the *SurVo Advanced User's Guide*.

### Configuring the NetGet

After creating a SurVo, select any post-survey action of Net Integration (with or without emailing results or storing in the ifbyphone database—see the *SurVo Advanced User's Guide* for additional information). Input the parameters defining the action that will be taken on your server via the SurVo Netget page. The values you select will build the URL that will be executed upon SurVo completion. Depending on the NetGet action you select, you will need to prepare the appropriate file or page on the server configured for the NetGet connection.

**Survo NetGet**

Configure the URL that will be used to submit SurVo results to your website/server and what action it should take with the results.

**NetGet Action:** Transmit data and get next action

**Submit Type:** GET

**Domain:** http://24.12.151.72

**Page:** netget\_test\_response.php

**Static Parameters:**

---

**SurVo Generated Parameters:**

**Parameter 1:** id = A1

**Parameter 2:** phone = Caller Id

**Parameter 3:** unique = Unique Id

**Parameter 4:** passthrough = Passthrough data

**Parameter 5:** action = A2

**Parameter 6:** timestamp = Timestamp

**Parameter 7:**

To remove a parameter just delete its contents or select the blank option from the drop down and hit update.  
To add another parameter after current ones are filled in, hit Save/Update and a new empty parameter field will be added to the list.

Current URL:  
http://24.12.151.72/netget\_test\_response.php?id=A1&phone=callerid&unique=unique\_id&passthrough=p\_t&action=A2&timestamp=timestamp

Click Save/Update to see updated URL

Save/Update

Save and Return to SurVo

Figure 4: SurVo NetGet Sample

### The NetGet Page

The following settings are available on the SurVo NetGet page (Figure 5):

- **NetGet Action** (select from the drop-down list)
  - Retrieve HTML/TEXT and read to user
  - Retrieve RSS and read to user
  - Transmit data and disconnect
  - Transmit data and get next action
- **Submit Type** (choose from either GET or POST submit formats)
- **Domain** (enter the complete URL or Web address of the site)
- **Page** (enter the complete filename, including directory designation)
- **Static Parameters** (enter parameters to be passed to the NetGet page)
- **SurVo Generated Parameters** (connect a user-defined local value with items in the drop-down list)
  - Caller ID
  - Unique ID
  - Timestamp
  - Passthrough Data

Configure the URL that will be used to submit SurVo results to your website/server and what action it should take with the results.

<b>NetGet Action:</b>	<input type="text" value="Retrieve HTML/TEXT and read to user"/>
<b>Submit Type:</b>	<input type="text" value="GET"/>
<b>Domain:</b>	<input type="text" value="http://"/>
<b>Page:</b>	<input type="text"/>
<b>Static Parameters:</b>	<input type="text"/>
<b>SurVo Generated Parameters:</b>	
<b>Parameter 1:</b>	<input type="text"/> = <input type="text"/>

To remove a parameter just delete its contents or select the blank option from the drop down and hit update.  
To add another parameter after current ones are filled in, hit Save/Update and a new empty parameter field will be added to the list.

Figure 5: SurVo NetGet Fields



#### WARNING: Restrictions

POST will not work if the page is redirected. Also, do not use the following static parameters in a NetGet URL: A## (where # is a number (e.g. A12)), callerid, unique\_id, sid and p\_t.

NetGet Actions

- Retrieve HTML/TEXT and read to user**  
 Transmit the data and retrieve the designated HTML or text to be read to the respondent. The HTML will be stripped of its tags so the text will give you a clear and more predictable response.
- Retrieve RSS and read to user**  
 Transmit the data and retrieve an RSS-formatted data to be presented to the respondent with navigation controls to move through the “articles.” You must use a GET when employing this option.
- Transmit data and disconnect**  
 Upon data transmission, the respondent will be disconnected from the call, unless the SurVo has already transferred the call or handed off to another building block.
- Transmit data and get next action**  
 Upon data transmission, return an XML-based file containing the next instruction.  
 Note: If doing a transfer or if the SurVo routed the caller to a building block (e.g. Voicemail, Virtual Receptionist, Find Me), data will be transmitted but the next action will be ignored.

Action App	Parameters	Value/Notes
survey	id	An existing survo_id of the next Survo to go to.
findme	id	An existing findme_id of the next Find Me to go to.
	nextaction	Action to take if final number on list does not answer or declines the call. Valid values are 1 – tell user no one is available and disconnect; 2 – transfer to voice mail; 3 – automatically accept call on last number tried; 7 – automatically accept call without intro.
	nextactionitem	Ifbyphone Building Block ID of a voice mailbox. Required if nextaction = 2.
	phone_list	A “   “ delimited list of phone numbers to try in the order specified.
vr	id	An existing mymenu_id of a Virtual Receptionist to go to.
voicemail	id	An existing vmail_id of a Voice Mailbox to go to.
hangup		Hangs up the call.

**The XML structure for an action statement:**

```

<action>
  <app>survo</app>
  <parameters>
    <id>75</id>
    <user_parameters>
      <param1>value1</param1>
      <param2>value2</param2>
    </user_parameters>
    <p_t>mypassthrough data</p_t>
  </parameters>
</action>

```

### Submit Type

For most NetGet actions, choose either a POST or GET command for transmission to your server. The “Retrieve RSS” action will do a GET regardless of this setting.

### Domain

Enter the domain of your server.

- An example for standard data transmission is <http://myserver.com>.
- An example using secure transmission is <https://myserver.com>.

### Page

Enter the filename on the server to receive the request, which may include multiple directory levels.

### Static Parameters

Enter parameters that do not come from the SurVo or as pass-through data. Keyword/value pairs should be inputted in URL GET format; each pair separated with an ampersand (&). A common use of this is to track a step in process (e.g. Step=Authorize) so your code knows what to do. See next page for an example.

### Dynamic Parameters

Enter the items from the SurVo that you want to pass back to your server. The parameter name that you wish to receive should be inputted on the left-hand side, while the SurVo variable should be selected from the corresponding drop-down list. Initially, there is one dynamic parameter field available. To add more to the NetGet URL, press the **Save/Update** button. When the page redisplay, another dynamic parameter will be available. Each time you do a Save/Update, the page will display the “current url” based on your current settings.

You may choose from the following parameters:

- **SurVo Answers** – Pass the answers from the SurVo by selecting A1 for the first answer, A2 for the second answer, and so on.
- **Caller Id** – The phone number of caller or, if outbound, the number called during this session.
- **Unique Id** – A unique reference ID assigned to this session by the ifbyphone system.
- **Timestamp**
- **Passthrough Data** – Data you submitted on a Click-to or in response to a previous NetGet. Think of this as a “hidden field” in your form.

### Example: NetGet

This simple example uses a SurVo to collect initial information, and then passes the results via a NetGet to another URL, which in turn processes the information to determine the next action to be taken. The information is then loaded into a “next action” XML structure and passed back to ifbyphone for further processing.

By following the steps outlined in the *Survo Advanced User’s Guide*, a SurVo is created with Net Integration to prompt for the respondent’s account information, where the first question of the SurVo requests his or her account number. Attached to the SurVo is a NetGet (Figure 6), which passes the account number to a specific Web site. Since the destination page will pass back the next action to be taken, the NetGet will be of type “Transmit data and get next action.”

**Survo NetGet**

Configure the URL that will be used to submit SurVo results to your website/server and what action it should take with the results.

**NetGet Action:**

**Submit Type:**

**Domain:**

**Page:**

**Static Parameters:**

---

**SurVo Generated Parameters:**

**Parameter 1:**  =

Figure 6: Sample NetGet

Assuming that a file named phone\_interface.php on your site <http://www.yourwebsite.com> will be handling the validation, the URL generated will look like this (Figure 7):

[https://yourwebsite.com/phone\\_interface.php?step=authorize&attempt\\_number=1&acct\\_number=A1](https://yourwebsite.com/phone_interface.php?step=authorize&attempt_number=1&acct_number=A1)

Current URL: [http://yourwebsite.com/phone\\_interface.php?step=authorize&attempt\\_number=1&acct\\_number=A1](http://yourwebsite.com/phone_interface.php?step=authorize&attempt_number=1&acct_number=A1)

Click Save/Update to see updated URL

---

Figure 7: Sample NetGet URL

A1 will be substituted during SurVo execution with the answer to the first question, the respondent's Account ID. The other parameters, which indicate the current step and number of validation attempts, are set up in the NetGet as static parameters.

**Pseudo-code for the destination URL**

```
//File: phone_interface.php
//Contains the logic for a customer validating an account number,
//triggered by various SurVo NetGets.

// Retrieve the values passed in the URL
$step = $_REQUEST['step'];
$attempt_number = $_REQUEST['attempt_number'];
$account_number = $_REQUEST['acct_number'];

// Take action based on the values.
switch($step){
  case 'authorize': // Validate the account number passed.
    $account_status = valid_account_number($account_number);
    // Test status returned by validation routine
    if ($account_status == 'valid') {
      // Build next action XML structure to pass designated SurVo id
      // for continued processing
      echo "<action>
        <app>survo</app>
        <parameters>
          <id>###</id>
          <p_t>acct|$account_number</p_t>
        </parameters>
      </action>";
    } else {
      // Build next action XML structure to pass designated SurVo id for invalid accounts
      // First check how many failed attempts
      if ($attempt_number <= 2) {
        // Less than 2 attempts, request another attempt
        //at entering the account id.
        $attempt_number++;
        echo "<action>
          <app>survo</app>
          <parameters>
            <id>###</id>
            <user_parameters>
              <acct_number>$account_number </acct_number>
            </user_parameters>
            <p_t>attempt|$attempt_number</p_t>
          </parameters>
        </action>";
      } else {
        // Too many failed attempts.
        // Return next action XML structure to execute
        // SurVo which transfers the call to Customer Service

        echo "<action>
          <app>survo</app>
          <parameters>
            <id>###</id>
          </parameters>
        </action>";
      }
    }
    break;
  default:
    ;
} // end switch
exit;
```

Trigger a Reminder phone call

You can employ this method to call a specific phone number with a reminder announcement. This API call is different than the Smart Click-to-Call API call.

The following URL is an example of a typical format for a Reminder call :

```
https://secure.ifbyphone.com/vome_widget_add.php?user_phone=8475552222&pin=1234&message_text=time%20to%20begin%20workout%20training&recipient_name=James&caller_name=my%20Reminder%20Service&M=12&D=28&Y=2007&H=4&N=30&A=PM&timezone=Central
```

The following table lists each of the parameters and expected values for the Reminder API call.

Application	Parameter	Value/Notes
Reminder	User_phone	Phone number registered to your account
	Pin	Personal Identification Number associated with your account
	Message_text	Message to be spoken to the recipient of the phone call.
	Recipient_name	Name to be spoken to personalize the message.
	Caller_name	Name to be spoken to identify the originator of the message.
	M	The Month of the year that the call should be placed
	D	The day of the Month M, that the call should be placed
	Y	The year of the date that the call should be placed
	H	The hour of the day that the call should be placed
	N	The minute of the hour H that the call should be placed
	A	Meridian (valid values AM , PM)
	Timezone	Timezone for the time of the call (valid values : Eastern, Central, Mountain, Pacific, Alaska, Hawaii)

Set a Schedule mode

You can employ this method to change the current mode of a schedule attached to your account. This API call is different than the Smart Click-to-Call API call.

To set a schedule mode use, the following API call to determine the available schedules, available modes for a schedule and finally set the mode for the schedule in question.

The following URL is an example of a typical format for a Reminder call :

```
http://www.ifbyphone.com/ibp_schedule_api.php?acct_id=9999&func=get_current_mode&usr_schedule_id=1234
```

The following table lists each of the parameters and expected values for the Reminder API call.

Application	Parameter	Value/Notes
Schedule	Acct_id	Your ifbyphone account identification number
	Func	Function request valid values "get_current_mode", "get_avail_modes", "set_mode"hold
	Usr_schedule_id	The identification number of a schedule assigned to your account.

The normal method for setting a schedule override consists of two steps. Determine the current mode for the specified schedule and then set the schedule to the mode required. The following example shows the calls for those steps.

This call returns the list of modes that are available for the schedule with id = 1234

```
http://www.ifbyphone.com/ibp_schedule_api.php?acct_id=9999&func=get_avail_modes&usr_schedule_id=1234
```

Returned information : Open|577||Closed|578||Lunch|579

This API call will return the current mode that the schedule associated with the ID 1234

```
http://www.ifbyphone.com/ibp_schedule_api.php?acct_id=9999&func=get_current_mode&usr_schedule_id=1234
```

Returned information : Closed|578

This API call will set the current mode for schedule 1234 to Open (value 577)

```
http://www.ifbyphone.com/ibp_schedule_api.php?acct_id=9999&func=set_mode_hold&usr_schedule_id=1234&usr_mode_id=577
```

Returned information : success

The mode of Open will remain in effect until such time as you access your account using the ifbyphone GUI or you use another API call to change the mode of the schedule to a different mode.

# 4

## Administrative API

*The Administrative API gives you different sets of methods, parameters, and values to perform various administrative tasks on the Ifbyphone system.*

The Ifbyphone Administrative API lets you perform a variety of administrative tasks on the Ifbyphone system, such as setting up voicemail boxes and pulling reports. Each request requires an action parameter which defines what type of object/item you are trying to work with, and what you want to do with it (method). The standard form is action=object.method, so for example action=vmail.createbox. The following tables provide the information on the methods supported for different objects and the required and optional parameters for those methods.

In addition to the action parameter, all API requests also require the api\_key for your account. You can find this key by going to the building block page under tools on the system.

A correct implementation of the API should look like the following (the parameters can be in any order):

[https://secure.ifbyphone.com/ibp\\_api.php?action=object.method&api\\_key=ahaajw.....78kjgh](https://secure.ifbyphone.com/ibp_api.php?action=object.method&api_key=ahaajw.....78kjgh)

All API requests will return an xml structured response. The first part of that structure will always contain a <result> tag and a <result\_description> tag. If the request failed you will get back a response like this:

```
<response>
  <result>failed</result>
  <result_description>Missing box name</result_description>
</response>
```

If it was successful you will get back a result of 'success' and the result\_description will be empty. This will be followed by additional tags with the information based on the action requests.

```
<response>
  <result>success</result>
  <result_description/>
  . . .
</response>
```

Vmail		
Method	Parameter	Value/Notes
<b>.createbox</b>	name	Required – the identifier for the mail box .
	pin	Required – must be 4 digits long.
	email_address	Optional – a basic check of valid email address formatting will be done. Used if we are sending vmails via email.
	send_email	Optional – but only will be used if email_address is provided, otherwise it will be set to false (0). Valid values are 0 or 1, however anything but a 1 will be treated as a 0.
	envelope	Optional – valid values are 0 or 1, however, anything but a 1 will be treated as a 0.
Returned if success: <pre>                     &lt;response&gt;                     &lt;result&gt;success&lt;/result&gt;                     &lt;result_description/&gt;                     &lt;box_id&gt;254&lt;/box_id&gt;                     &lt;/response&gt;                     </pre>		

<b>.getmessages</b>	box_id	Required - the ID of the vmail box
	greetings	Optional -
Returned if success: <pre>                     &lt;response&gt;                     &lt;result&gt;success&lt;/result&gt;                     &lt;result_description/&gt;                     &lt;data&gt;                     &lt;message&gt;                     &lt;url&gt;                     (path to file)                     &lt;/url&gt;                     &lt;envelope&gt;5555551234 on Mon January 14, 2008, 6:23 pm&lt;/envelope&gt;                     &lt;/message&gt;                     &lt;message&gt;                     &lt;url&gt;                     (path to file)                     &lt;/url&gt;                     &lt;envelope&gt;5555551234 on Tue January 15, 2008, 3:45 pm&lt;/envelope&gt;                     &lt;/message&gt;                     &lt;/data&gt;                     &lt;/response&gt;                     </pre>		

<b>.recordgreeting</b>	box_id	Required – the mailbox ID for which the greeting is being recorded.
	recording_type	Required – which recording type. Valid options are: greeting, vacation.
	phone_to_call	Required – the phone number to call to record the greeting.
Returned if success: <pre>                     &lt;response&gt;                     &lt;result&gt;Success&lt;/result&gt;                     &lt;result_description&gt;Call Attempted&lt;/result_description&gt;                     &lt;/response&gt;                     </pre>		

Report		
Method	Parameter	Value/Notes
<b>.clickto</b>	start_date	Required –date to begin running the query. The beginning time will be midnight on this day. Date needs to be formatted YYYYMMDD
	end_date	Required – date the query will end. The ending time will be 23:59 on this day.
	click_id	Optional – a specific Smart Click-to-Call building block ID.
	ref	Optional – a specific value withing the “ref” return value (example: for a report on all calls with ref = “home page”
<p>Returned if success:</p> <pre> &lt;data&gt;   &lt;record&gt;     &lt;Caller_ID&gt;8475551234&lt;/Caller_ID&gt;     &lt;Name&gt;Phone-me-now&lt;/Name&gt;     &lt;Page/&gt;     &lt;Ref&gt;12345678&lt;/Ref&gt;     &lt;Click_Description&gt;URL based&lt;/Click_Description&gt;     &lt;Click_Date_Time&gt;Mar 03 2008 4:25 PM&lt;/Click_Date_Time&gt;     &lt;Number_Connected&gt;8475554321&lt;/Number_Connected&gt;     &lt;Call_Duration/&gt;     &lt;Referral_Keywords/&gt;     &lt;Referral_Domain/&gt;     &lt;IP_Address&gt;192.168.5.235&lt;/IP_Address&gt;     &lt;Call_1_Result&gt;Phone Busy&lt;/Call_1_Result&gt;     &lt;Call_2_Result/&gt;   &lt;/record&gt; &lt;/data&gt; </pre>		

SurVo		
Method	Parameter	Value/Notes
<b>.get_recording</b>	api_key	Required – the account’s API key
	format	Required – what audio format to stream – wav or mp3
	question	Required – the question number you are requesting (1,2,3...)
	sample_rate	Optional – sets the sample rate of the file that is streamed. Defaults to 8000, can be set to 11000, 22050, 44100
	survo_id	Required – the SurVo id of the SurVo you want
	unique_id	Required – the unique id of this SurVo’s response
<p>Example call:  <a href="https://secure.ifbyphone.com/ibp_api.php?api_key=e341a947698451284edd18ac8ff05e2ac6d96939&amp;action=survo.get_recording&amp;survo_id=475&amp;unique_id=2921&amp;question=1&amp;format=mp3">https://secure.ifbyphone.com/ibp_api.php?api_key=e341a947698451284edd18ac8ff05e2ac6d96939&amp;action=survo.get_recording&amp;survo_id=475&amp;unique_id=2921&amp;question=1&amp;format=mp3</a></p>		

Verify-Me-Now		
Method	Parameter	Value/Notes
<b>.verify</b>	api_key	Required – the account's API key
	verify_id	Required – the verify id of the Verify-Me-Now you want
	phone_number	Required – the phone number to call
	pin	Required – the PIN number that the user must enter to be verified. This can be up to 10-digits.
<p>Example call:  <a href="https://secure.ifbyphone.com/ibp_api.php?api_key=e341a947698451284edd18ac8ff05e2ac6d96939&amp;action=verifymenow.verify&amp;verify_id=231&amp;phone_number=8475551212&amp;pin=1234">https://secure.ifbyphone.com/ibp_api.php?api_key=e341a947698451284edd18ac8ff05e2ac6d96939&amp;action=verifymenow.verify&amp;verify_id=231&amp;phone_number=8475551212&amp;pin=1234</a></p>		
<p>Returned if success:</p> <pre>&lt;response&gt;   &lt;result&gt;verified&lt;/result&gt;   &lt;data&gt;     &lt;sid&gt;0805089531254086&lt;/sid&gt;   &lt;/data&gt; &lt;/response&gt;</pre>		

Method	Parameter	Value/Notes
<b>.get_recording</b>	api_key	Required – the account's API key
	format	Required – what audio format to stream – wav or mp3
	sample_rate	Optional – sets the sample rate of the file that is streamed. Defaults to 8000, can be set to 11000, 22050, 44100
	verify_id	Required – the verify id of the Verify-Me-Now you want
	sid	Required – the unique call id of the call (returned by the .verify method)
<p>Example call:  <a href="https://secure.ifbyphone.com/ibp_api.php?api_key=e341a947698451284edd18ac8ff05e2ac6d96939&amp;action=verifymenow.get_recording&amp;verify_id=231&amp;sid=0805089531254086&amp;format=mp3">https://secure.ifbyphone.com/ibp_api.php?api_key=e341a947698451284edd18ac8ff05e2ac6d96939&amp;action=verifymenow.get_recording&amp;verify_id=231&amp;sid=0805089531254086&amp;format=mp3</a></p>		

Findme		
Method	Parameter	Value/Notes
<b>.add_number</b>	findme_id	Required – the ID of the Find Me to add the phone number to
	phone_number	Required – the phone number to add.
<p>Returned if success:</p> <pre>&lt;response&gt;   &lt;result&gt;success&lt;/result&gt;   &lt;result_description&gt;Phone number 5555555555 added to findme 111&lt;/result_description&gt; &lt;/response&gt;</pre>		

<b>.create</b>	distribution_type	Optional – Valid values are 1 or 2. 1 to randomize find me list, 2 to use round robin to select number. Default is normal sequencing.
	findme_action	Optional – Action to be performed at the end of the list. See appendix for list of valid find me actions.
	findme_action_parameter	Optional – If necessary, building block ID for find me action. See appendix to check if this is required.
	list_type	Optional – 1 if individual find me, 2 for customer service hunt group. 1 is default.
	loop_count	Optional – Number of times to repeat find me list. Valid value is a number between 1 and 5. Default is 1.
	name	Required – the name of the Find Me
	record	Optional – valid values are 0 or 1. Anything but a 1 will be treated as a 0.
	use_screen	Optional – Valid values are 0 or 1. Anything but a 1 will be treated as a 0.
	whisper_phrase	Optional – phrase to be whispered if screening is not enabled. Default is blank

Returned if success:

```
<response>
  <result>success</result>
  <result_description/>
  <findme_id>111</findme_id>
</response>
```

<b>.delete</b>	findme_id	Required – the ID of the Find Me to delete.
----------------	-----------	---

Returned if success:

```
<response>
  <result>success</result>
  <result_description>findme 111 has been deleted</result_description>
</response>
```

<b>.delete_number</b>	findme_id	Required – the ID of the Find Me to remove the phone number from.
	findme_number_id	Required – the ID of the phone number to remove (IMPORTANT: this is the ID of the phone number, not the phone number itself).

Returned if success:

```
<response>
  <result>success</result>
  <result_description>Findme number id 999 removed from findme 111</result_description>
</response>
```

<b>.delete_recorded_call</b>	findme_id	Required – the ID of the Find Me to delete.
	recording_name	Required – the file name of the recording to delete.

Returned if success:

```
<response>
  <result>success</result>
  <result_description>Findme recording has been deleted</result_description>
</response>
```

<b>.get_findme_list</b>	no parameters required
<p>Returned if success:</p> <pre>&lt;response&gt;   &lt;result&gt;success&lt;/result&gt;   &lt;result_description/&gt;   &lt;data&gt;     &lt;findme&gt;       &lt;id&gt;999&lt;/id&gt;       &lt;name&gt;Name for the Findme&lt;/name&gt;     &lt;/findme&gt;   &lt;/data&gt; &lt;/response&gt;</pre>	

<b>.get_phone_list</b>	findme_id	Required – the ID of the Find Me to retrieve list of numbers from.
<p>Returned if success:</p> <pre>&lt;response&gt;   &lt;result&gt;success&lt;/result&gt;   &lt;result_description/&gt;   &lt;data&gt;     &lt;phone&gt;       &lt;findme_number_id&gt;999&lt;/findme_number_id&gt;       &lt;number&gt;5555555555&lt;/number&gt;       &lt;priority&gt;1&lt;/priority&gt;     &lt;/phone&gt;   &lt;/data&gt; &lt;/response&gt;</pre>		

<b>.get_recorded_calls_list</b>	findme_id	Optional – if specified only audio recordings for a specific find me will be returned.
<p>Returned if success:</p> <pre>&lt;response&gt;   &lt;result&gt;success&lt;/result&gt;   &lt;result_description/&gt;   &lt;data&gt;     &lt;findme&gt;       &lt;findme_id&gt;111&lt;/findme_id&gt;       &lt;recorded_call&gt;         &lt;url&gt;           (path to file)         &lt;/url&gt;         &lt;date&gt;04/02/2008 09:23:43&lt;/date&gt;         &lt;callerid&gt;5555555555&lt;/callerid&gt;         &lt;connected_number&gt;5555555555&lt;/connected_number&gt;       &lt;/recorded_call&gt;     &lt;/findme&gt;   &lt;/data&gt; &lt;/response&gt;</pre>		

<b>.update_list_settings</b>	distribution_type	Optional – Valid values are 1 or 2. 1 to randomize find me list, 2 to use round robin to select number. Default is normal sequencing.
	findme_action	Optional – Action to be performed at the end of the list. See appendix for list of valid find me actions.
	findme_action_parameter	Optional – If necessary, building block ID for find me action. See appendix to check if this is required.
	list_type	Optional – 1 if individual find me, 2 for customer service hunt group. 1 is default.
	loop_count	Optional – Number of times to repeat find me list. Valid value is a number between 1 and 5. Default is 1.
	name	Required – the name of the Find Me
	record	Optional – valid values are 0 or 1. Anything but a 1 will be treated as a 0.
	use_screen	Optional – Valid values are 0 or 1. Anything but a 1 will be treated as a 0.
	whisper_phrase	Optional – phrase to be whispered if screening is not enabled. Default is blank
<p>Returned if success:</p> <pre>&lt;response&gt; &lt;result&gt;success&lt;/result&gt; &lt;result_description&gt;findme options for findme 259 have been updated&lt;/result_description&gt; &lt;/response&gt;</pre>		

<b>Appendix: Find Me Actions</b>		
<b>Action</b>	<b>Description</b>	<b>Action Parameter</b>
1	Tell user no one is available and disconnect	N/A
2	Transfer to Voice Mail	Voice Mail ID
3	Automatically accept call on last number tried	N/A
4	Route to a Virtual Receptionist	Virtual Receptionist ID
5	Route to a different Find Me list	Find Me ID
6	Route to a SurVo	SurVo ID
7	Automatically accept call without intro	N/A

General		
Method	Parameter	Value/Notes
<b>.login</b>	username	Required – username
	password	Required – password
<p>Example call:  <a href="https://secure.ifbyphone.com/ibp_api.php?action=general.login&amp;username=5102847452&amp;password=1234">https://secure.ifbyphone.com/ibp_api.php?action=general.login&amp;username=5102847452&amp;password=1234</a></p>		
<p>Returned if success:</p> <pre>&lt;response&gt;   &lt;result&gt;e341a947698459184edd18ac80905e2ac6d96939&lt;/result&gt; &lt;/response&gt;</pre> <p>Returned if failure:</p> <pre>&lt;response&gt;   &lt;result&gt;invalid&lt;/result&gt; &lt;/response&gt;</pre>		

Broadcast		
Method	Parameter	Value/Notes
<b>.create</b>	api_key	Required – the account's API key
	name	Required – name of the new broadcast
	phone_number_list	Required – bar delimited list of phone numbers to call
	recording_phone_number	Optional – this number will be called to record message. If this is used, then audio_file is not used.
	audio_file	Optional – audio file to play as message.
	timestamp	Optional – when should broadcast begin (Y-m-d H:i:s). If blank, and schedule=1, will begin broadcast immediately after recording message
	schedule	Optional – if 1, then do broadcast now. If null or 0, broadcast will be saved, but not sent.
<p>Example call:  <a href="https://secure.ifbyphone.com/ibp_api.php?api_key=e341a947698459184edd18ac80905e2ac6d96939&amp;action=broadcast.create&amp;name=broadcast1&amp;phone_number_list=8475551212%7C8475551414&amp;recording_phone_number=8475551234&amp;timestamp=">https://secure.ifbyphone.com/ibp_api.php?api_key=e341a947698459184edd18ac80905e2ac6d96939&amp;action=broadcast.create&amp;name=broadcast1&amp;phone_number_list=8475551212%7C8475551414&amp;recording_phone_number=8475551234&amp;timestamp=</a></p> <p>This can be a POST instead of a GET. If using audio_file, must be a POST.</p>		
<p>Returned if success:</p> <pre>&lt;response&gt;   &lt;result&gt;success&lt;/result&gt;   &lt;result_description&gt;1 Voice Broadcast(s) scheduled&lt;/result_description&gt;   &lt;audio_dialog_id&gt;372&lt;/audio_dialog_id&gt;   &lt;basic_broadcast_id&gt;150&lt;/basic_broadcast_id&gt; &lt;/response&gt;</pre>		

<b>.results</b>	api_key	Required – the account's API key
	basic_broadcast_id	Required – broadcast to get results for
<p>Example call:  <a href="https://secure.ifbyphone.com/ibp_api.php?api_key=e341a947698459184edd18ac80905e2ac6d96939&amp;action=broadcast.results&amp;basic_broadcast_id=14">https://secure.ifbyphone.com/ibp_api.php?api_key=e341a947698459184edd18ac80905e2ac6d96939&amp;action=broadcast.results&amp;basic_broadcast_id=14</a></p>		
<p>Returned if success:</p> <pre>&lt;response&gt;   &lt;result&gt;success&lt;/result&gt;   &lt;call&gt;     &lt;call_time_scheduled&gt;2008-05-29 13:05:03&lt;/call_time_scheduled&gt;     &lt;call_time_actual&gt;2008-05-29 13:05:03&lt;/call_time_actual&gt;     &lt;call_delay&gt;00:00:00&lt;/call_delay&gt;     &lt;number_called&gt;8476766624&lt;/number_called&gt;     &lt;call_result&gt;Scheduled&lt;/call_result&gt;   &lt;/call&gt; &lt;/response&gt;</pre> <p>call_result could be: Scheduled, Live Person, Answering Machine, HANGUP, or NOTSURE.</p>		

<b>.stop</b>	api_key	Required – the account's API key
	Basic_broadcast_id	Required – broadcast to stop
<p>Example call:  <a href="https://secure.ifbyphone.com/ibp_api.php?api_key=e341a947698459184edd18ac80905e2ac6d96939&amp;action=broadcast.stop&amp;basic_broadcast_id=14">https://secure.ifbyphone.com/ibp_api.php?api_key=e341a947698459184edd18ac80905e2ac6d96939&amp;action=broadcast.stop&amp;basic_broadcast_id=14</a></p>		
<p>Returned if success:</p> <pre>&lt;response&gt;   &lt;result&gt;success&lt;/result&gt;   &lt;result_description&gt;2 Basic Broadcast(s) Stopped&lt;/result_description&gt; &lt;/response&gt;</pre>		



## Ifbyphone Glossary

*The purpose of the glossary is to assist you by defining commonly-used ifbyphone terms.*

### **Auto attendant**

Another term for Interactive Voice Response (IVR). Essentially refers to an automated telephone answering system that routes calls.

### **Building Block**

Ifbyphone proprietary code that allows you to build a complex voice application that is accessible via the Web, Email or telephone. Smart Click-to-Call and all destinations are building blocks.

### **Click to Call (CTC)**

Immediately connect a user to a registered business telephone number. Click to Call can be easily set up by registering phone numbers—in addition to the toll free number given to you at sign up—within the Registered Numbers utility. Click to Call can then be configured as a destination for Smart Click-to-Call, enabling Web site visitors to click a button and immediately speak with a customer service representative or other call recipient.

### **Destinations**

Destinations are ifbyphone services that can be configured for Smart Click-to-Call access. Through Smart Click-to-Call, customers can click a link, type in their phone number, and connect immediately to a destination.

### **Destination Configurations**

Destination Configurations allow you to setup a destination for use with a specific Smart Click-to-Call.

### **Interactive Voice Response (IVR)**

A computerized system that allows a telephone caller to select options from a voice menu and interact with the computer phone system. IVR systems use Dual Tone, Multi-Frequency (DTMF) signals (entered from the telephone keypad) and natural language speech recognition to interpret the caller's response to IVR prompts.

**Find Me**

An Ifbyphone business service and destination, Find Me can be configured to maintain a prioritized list of phone numbers where a call recipient may be reached (e.g. desk or cell phone, pager, etc.). If a call recipient cannot be located at any of the phone numbers listed, the call can be redirected to their voice mailbox.

**Modes**

Blocks of time established within a specific day when configuring a schedule. Modes are fully customizable, but are commonly used to represent a workplace's open, closed, lunch, and after hours. Once modes have been set within a schedule, Smart Click-to-Call and Virtual Receptionist can be configured to exhibit different IVR behavior for a specific mode.

**Net Integration**

Communication between a SurVo and Web-based server. Net integration may be carried out to retrieve or post information and to perform a function like user account validation. Typically net integration is performed when a company is utilizing ifbyphone business services to add voice to an existing system.

**Schedule**

An Ifbyphone tool that allows a user to specify the day-to-day operation of a workplace with start and finish times. Schedules are configured for ifbyphone business services, including Smart Click-to-Call and Virtual Receptionist. Within a schedule, it is possible to specify a desired time zone, as well as modes.

**Software as a Service (SaaS)**

A software model employed by vendors who both develop and operate network-based applications for use by customers over the internet. SaaS is typically a low-cost way for businesses to obtain the same benefits of commercially licensed, internally-operated software, without the associated complexity and high initial cost.

**Smart Click-to-Call**

An Ifbyphone service that is highly flexible in customization and scalability, Smart Click-to-Call enables routed communication between customers, business services and contacts via the Web or Email. Through Smart Click-to-Call, customers can click a link, type in their phone number, and connect immediately to a company's configured services (commonly referred to as "destinations" in this documentation). Ifbyphone offers the following services as Smart Click-to-Call destinations: a custom Virtual Receptionist, voice survey, voice mailbox, a Find Me auto-dial telephone list to locate an individual, and Click to Call for immediate telephony connection. Smart Click-to-Call can be configured based on the day and/or time of day.

**SurVo**

See Voice Survey Form (SurVo)

**Virtual Receptionist**

An ifbyphone business service and destination, Virtual Receptionist can be configured to provide menus for callers to route incoming calls. An automated phone environment, the virtual receptionist can transfer callers to other numbers/extensions, forward callers to voicemail boxes, initiate a Find Me list or surveys, play pre-recorded messages, etc.

**Voice Form**

Similar to a Web form—with voice recognition capabilities, a voice form is a dialog between a user and the system configured to provide or collect information. At the completion of a voice form, the information collected can be emailed, saved to a web site or transmitted for further processing.

**Voice Survey Form (SurVo)**

An ifbyphone business service and destination, Voice Survey Forms (SurVo) are used to create dialogs over the telephone between callers and the system, much like a Web form with voice recognition capabilities (see Voice Form). A SurVo can be used to conduct employment screens, provide a customer satisfaction survey, route a call based on a user's answers to a series of questions, etc. SurVos have the potential to be invoked in response to a Smart Click-to-Call, an inbound call, an outbound call, a scheduled call, or an API request from a Web site.

**VoiceXML**

W3C's standard XML format for specifying interactive voice dialogues between a human and a computer. VoiceXML allows voice applications to be developed and deployed for visual applications. Analogous to HTML interpretation by a visual Web browser, VoiceXML documents are interpreted by a voice browser. VoiceXML has tags that instruct the voice browser to provide speech synthesis, automatic speech recognition, dialog management and soundfile playback.

**VUI (Voice User Integration)**

Describes the interaction with computers through a voice and/or speech platform to initiate an automated service or process.

**World Wide Web Consortium (W3C)**

An international consortium, hosted by MIT, which develops interoperable technologies (standards, protocols, software and tools) for the Web. W3C's mission is to promote the evolution of the Web in a single direction, rather than into splintered factions.