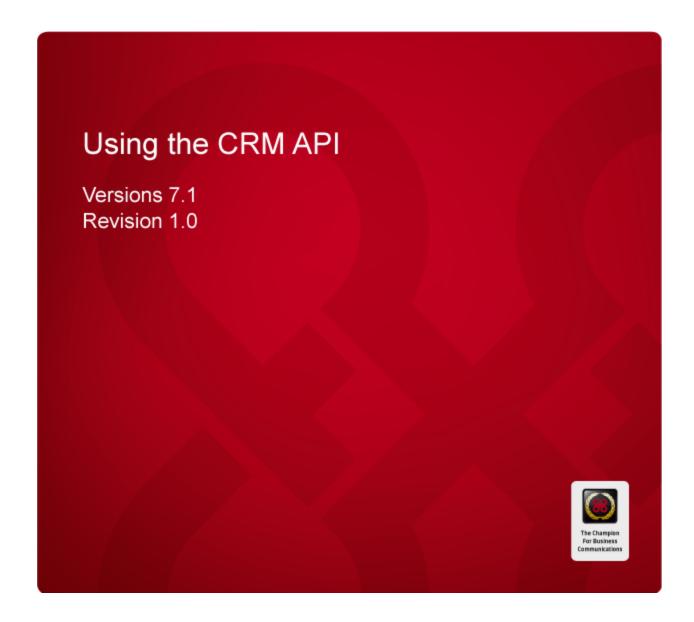


Virtual Contact Center



Copyright © 2012, 8x8, Inc. All rights reserved.

This document is provided for information purposes only and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

8x8® is a registered trademark of 8x8, Inc.

Microsoft® Internet Explorer is a trademark or a registered trademark of Microsoft Corporation in the United States and other countries.

All other trademarks, service marks, registered trademarks, or registered service marks are the property of their respective owner/s. All other brands and/or product names are the trademarks (or registered trademarks) and property of their respective owner/s.

Preface

Use this document to learn how to use the CRM API to request and submit data to and from the Internal CRM.

8x8, Inc 2125, O' Nel Drive San Jose, CA 94085 USA

SALES: 1.877.725.2621 SUPPORT: 1.866.975.2273

vccsales@8x8.com www.8x8.com

Contents

Preface	i
Introduction to the CRM API	1
Customer	1
Case	1
Follow-up.	_
Task	2
Objects	3
Actions.	3
Using the CRM API	5
Configuring CRM API Access.	5
Incoming Request	θ
List of Parameters	7
Handling Special Characters in CRM API Requests.	8
Responses to Queries.	<u>c</u>
Examples of Queries and Responses.	10
Manipulating Customer Objects	13
Add Customer.	13
Mandatory and Optional Parameters.	14
Reply from CRM to Add Customer.	16
Modify Customer	17
List of Mandatory and Optional Parameters.	17
Reply from CRM to Modify Customer	19
Get Customer Details.	20
Examples of Get Customer Operations.	20
List of Parameters	2 3
Reply from CRM to Get Customer Details.	24
Manipulating the Case Object.	26
Add Case	26
List of Mandatory and Optional Parameters.	27
Reply from CRM to Add Case.	29
Modify Case	
List of Mandatory and Optional Parameters.	
Reply from CRM to Modify Case	
Get Case.	
Tags that accept substring values	
Tags that accept multiple parameters	2.4

iii

Tags that can be paired for "range" queries and creation dates.	34
Examples of Get Case Operations.	35
List of Parameters.	42
Reply from CRM to Get Case.	44
Manipulating the Follow-up Object	46
Add Followup.	
List of Mandatory and Optional Parameters.	_
Reply from CRM to Add Followup.	
Get Followup.	_
List of Parameters.	
Reply from CRM to Get Followup.	
Manipulating the Task Object	
Manipulating the Task Object.	53
Add Task	
List of Mandatory and Optional Parameters.	
Reply from CRM to Add a Task	
Modify Task	
List of Mandatory and Optional Parameters.	
Reply from CRM to Modify a Task	59
Get Task	
Tags that accept substring values.	60
Tags that accept pre-defined values.	
Tags that can be paired for "range" queries.	
Examples of Get Queries.	
List of Parameters.	
Reply from CRM to Get Task	69
Manipulating the FAQCATEGORY Object	
Add FAQCATEGORY.	
Mandatory and Optional Parameters.	
Reply from CRM to Add FAQCATEGORY.	
Modify FAQCATEGORY	
Summary of Mandatory and Optional Parameters.	_
Reply from CRM to Modify FAQCATEGORY.	
Get FAQCATEGORY.	
Reply from CRM to Get FAQCATEGORY.	
Delete FAQCATEGORY. Mandatory and Optional Parameters.	
Reply from CRM to Delete FAQCATEGORY	/5
Manipulating the FAQ Object.	
Add FAO	76

Summary of Mandatory and Optional Parameters.	77
Reply from CRM to Add FAQ	78
Modify FAQ.	
Summary of Mandatory and Optional Parameter.	78
Reply from CRM to Modify FAQ	79
Get FAQ	80
List of Parameters.	81
Reply from CRM to Get FAQ	81
Delete FAQ	81
Summary of Parameters.	82
Reply from CRM to Delete FAQ	82
March July of the Control Publishers of Web as	00
Manipulating the Custom Field Objects and Values	
Get Custom Fields. List of Parameters.	
Reply from CRM to Get Custom Fields.	
Modify Custom Field Values. List of Mandatory and Optional Parameters.	
Reply from the CRM to Modify Custom Field Values.	
Reply from the Chivi to Woully Custom Field Values.	00
Listing the Attributes Values	87
List COUNTRY Attributes.	87
Reply from CRM to List COUNTRY.	87
Listing Tenant Configuration Attributes	QQ
List Agent GROUPS.	
Reply to List Agent GROUPS.	
List AGENTS.	
Reply to List AGENTS.	
List FAQ Categories.	
Reply to list FAQ Category.	
hepty to list mad category.	
Historical Statistics	91
Error Cases	92
GLOSSARY	95

Introduction to the CRM API

The Contactual CRM provides a model for managing your contact center's interactions with customers. The CRM stores your contact center's customer, case, and follow-up data. To manage customer relationships, the CRM provides the following hierarchy of standard objects:

- Customer
- Case
- Follow-up
- Task

Customer

A customer object allows you to create customer instances with unique Account numbers. A customer record stores information such as name, address, phone numbers, and offers the capability to customize the customer object with custom fields.

Case

A case object describes customer's feedback, queries, or issues. You can create cases to systematically track and solve customer's issues. You can quickly create, update, and view cases. You may indicate the status of a case as open or closed. The standard definition of a case allows you add additional attributes to a case by stating the status, visibility, severity, priority.

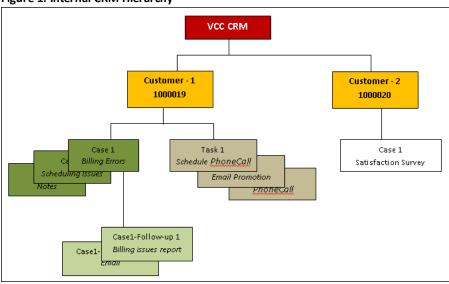
For example, a support agent at AcmeJets creates a case when a customer calls in reporting scheduling issues with private jets. The support agent captures and records the customer reported issue by creating a case.

Follow-up

A case may have multiple follow-ups before it is resolved. Each communication regarding the case from its creation until it is closed may be recorded as individual follow-up instances. This may include notes as well as email communications regarding the case. The Follow-up object allows you to track how a case is resolved by creating multiple follow-up records .

For example, When an AcmeJets customer reports scheduling issues, the support agent communicates with the Scheduling Supervisor to resolve the problem. All these communications by phone or email are transcribed as follow-up records.

Figure 1: Internal CRM Hierarchy



Task

A Task refers to a call, an email, a meeting, a chat or any other type of contact made with a customer. A Task organizes and helps you track all interactions associated with a customer. The task object serves to create, and manage all tasks you plan to perform or have performed, such as making calls or sending mails. For example, you can create a task to schedule a phone call to a customer.

This document describes how to use the CRM API component of the Contactual Integration Suite (formerly referred to as the Contactual WAPI). The CRM API component of the Integration Suite enables you to develop external programs that access the following Virtual Contact Center data:

- Internal CRM Customer, Case, Followup, and Task objects
- Internal CRM custom fields data for Customer, Case, Followup, and Task objects
- FAQ category and FAQ answer data

The CRM API uses HTTP or HTTPS to transmit requests and return results formatted as Extensible Mark-up Language (XML).

Objects

- Customer
- Case
- Follow up
- Task
- FAQ Category
- Custom Fields
- FAQ

Actions

- Add
- Modify
- Get
- Delete (FAQ Category and FAQ only)
- List

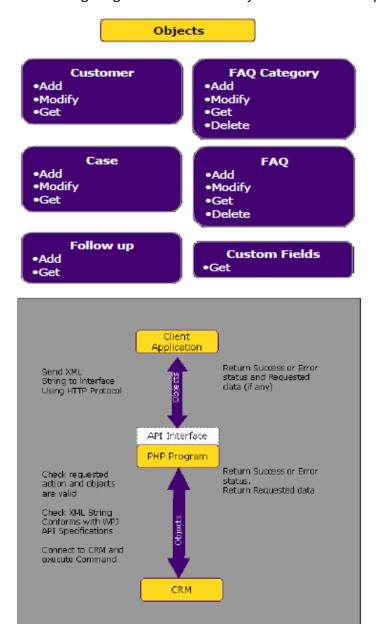
All actions executed through CRM API update the CRM database through data insertion, modification and retrieval.

The CRM API:

- Packages requests, and receives responses formatted as XML.
- Transmits and receives XML requests and responses via HTTP or HTTPS.

XML-based CRM API actions must be made by an authenticated user login and password. For information about configuring CRM API access to the Internal CRM, see the Contactual Configuration Manager Guide.

The following diagram outlines the object and CRM API application flow:



Using the CRM API

The API Interface accepts XML packets from external applications. The requests are sent via HTTP protocol. The interface expects a variable entitled "xml_query" which contains the xml packet. Once the API Interface receives a request, it processes the query as follows:

- 1. API Interface checks for an xml query variable.
- 2. XML string is extracted from the variable.
- 3. XML string is validated and parsed.
- 4. Data and command in XML string are validated.
- 5. Action is executed.
- 6. Error or success status is returned.
- 7. Requested data, if any, is returned.

The access point to the interface varies based on the platform your tenant is hosted on:

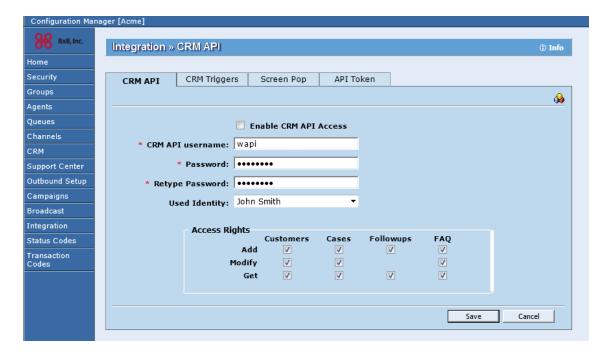
- For p5/p6 https://na1.mycontactual.com/WAPI/wapi.php
- For p7/p8 https://na2.mycontactual.com/WAPI/wapi.php
- For p9/p10 https://na3.mycontactual.com/WAPI/wapi.php
- For p11/p12 https://na4.mycontactual.com/WAPI/wapi.php

Configuring CRM API Access

Before using CRM API, you must enable and set it up in the Configuration Manager to define the Integration Suite user and configure the Integration Suite's permissions.

To configure CRM API access:

- 1. Login to Configuration Manager.
- 2. Navigate to **Integration** tab.



- 3. Check the Enable CRM API Access.
- 4. Enter a CRM API Username to access the API.

Note: Username length cannot exceed 10 characters.

5. Enter a Password to authenticate user access.

Note: Length of the password has to be between 3 and 15 characters.

- 6. Confirm your password by retyping.
- 7. **Used Identity:** Select an agent name whose identity will be used when accessing data thru the CRM API.
- 8. **Access Rights:** Specify data access rights to individual components allowed through the use of CRM API.

Incoming Request

XML requests to the CRM API must be submitted as a standard HTTP request on Port 80.

Example 1 illustrates an HTTP request.

Example 1: Example HTML containing HTTP request to Contactual XML API

```
In Example 1, WAPI_COMMAND specifies an XML request of the form
<COMMAND OBJECT="object_type" ACTION="action_type">
<TAG>param value</TAG>
```

Example 2: Example HTML containing HTTP request to Contactual XML API

All parameters included in WAPI COMMAND must contain a value.

List of Parameters

Table 1summarizes the parameters used in Example 2.

Table 1: Summary of parameters used in Example 2

tenant_name	Name of the tenant as defined in the console management.
wapi_username	The login for the agent as defined in the configuration manager
wapi_password	The password associated to the wapi_username.
object_type	Type of object manipulated. Valid values include CUSTOMER, CASE, FOLLOWUP, FAQ, FAQCATEGORY, and CUSTOMFIELD.
action_type	Action to perform on manipulated object. This could be any of the following values (depending on the type of the object): ADD, MODIFY, GET, LIST or DELETE.
param_value	Value given to the parameter defined by TAG.

Note: If XML request contains one or more quote characters (') they should be encoded as &rsquot;

Handling Special Characters in CRM API Requests

To send a request to the CRM API, you use HTTP (or HTTPS) to transmit the request formatted as XML. The XML-formatted request may include strings that include characters that have special meanings to HTTP.

For that reason, when sending a request to the CRM API that contains special characters, you must:

- Use the XML CDATA construct to contain the reserved characters
- Use both HTTP and XML "escape" encoding as a substitute for the reserved character
- Use the reserved characters to perform their special functions in CRM API searches

Table 2 lists common HTTP control characters and describes how to use those characters in CRM API requests.

Table 2: List of CRM API special characters

Character	Replace With	CRM API request encoding
"	Blank	No special encoding required.
1	Blank	Do not include the apostrophe character in CRM API requests.
		The apostrophe character is the Internal CRM database's string delimiter character.
~	Blank	No special encoding required.
!	Blank	No special encoding required.
#	No.	No special encoding required.
\$	Blank	No special encoding required.
%	Percent	The percent character is the CRM API wildcard character. For information about using Wildcard in CRM API requests, see <i>Tags</i> that accept substring values, on page 34, and <i>Examples of Get Case Operations</i> , on page 35.
٨	Blank	No special encoding required.
&	and	&
*	Asterisk	No special encoding required.
<	Blank	<
>	Blank	No special encoding required.

Responses to Queries

See "General form of a Internal CRM response" on page 9 illustrates a response from the Internal CRM.

Example 3: General form of a Internal CRM response

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<WAPI>
<REPLY STATUS= "error_code" ERROR_STR= "error_string">
<TAG> result_value</TAG>
</REPLY>
</WAPI>
```

In Example 3:

error_string contains a message that describes the reason the query was unsuccessful.

• result value contains the value returned for the parameter defined by TAG.

If the request was unsuccessful:

- If the request succeeds, error code = 0 (zero) and error string ="".
- If the request fails, error code="-1" and error string="".

See Error Cases, on page 92 for a list of errors in a unsuccessful reply.

If the reply contains a list of ITEMs:

- Each ITEM in the returned list is contained within XML begin and end tags of the general form <ITEM> and </ITEM>.
- Items for which no data is available are contained with empty tags are of the general form <TAG></TAG> or <TAG />.

Examples of Queries and Responses

Example 4 illustrates HTML that queries the Internal CRM for all customer records where first name is bill.

Example 4: HTML query for customer records that contain first name equal to "bill".

Example 5 illustrates Java code that queries the Internal CRM for all customer records that contain the first name equal to bill.

Example 5: Java example that retrieves all customer records with first name equal to "bill"

```
package WapiAPI;
```

```
import java.net.*;
import java.io.*;
* Title: Using Contactual CRM API
* Description:
* Copyright: Copyright (c) 2001
* Company: Contactual *
* @version 1.0
public class usePhp {
 public usePhp() {
public static void main(String[] args) {
int c;
try {
/*connect to wapi.php*/
URL source = new URL("http://na4.mycontactual.net/WAPI/wapi.php") ;
URLConnection conn;
conn = source.openConnection();
conn.setDoOutput(true);
/*create xml query string*/
String xmlString = "xml query=";
 xmlString +="<wapi>";
 xmlString +="<tenant>mandrake</tenant><username>wapi</username>";
 xmlString +="<password>wapi101</password><command object=\"customer\"
action=\"get\">";
 xmlString +="<firstname>bill</firstname></command>";
 xmlString +="</wapi>";
/*write data to connection*/
PrintWriter out = new PrintWriter(conn.getOutputStream());
out.println(xmlString);
```

Example 6 illustrates the response to the example queries in Example 4 or Example 5.

Example 6: Example response to guery for Internal CRM records with first name equal to Bill

```
<?xml version="1. 0" encoding="ISO-8859-1"?>
<WAPI>
<REPLY STATUS=0 ERROR STR="">
<ITEM>
<FIRSTNAME>Bill
<LASTNAME>Gates
<EMAIL>bill.gates@microsoft.com</EMAIL>
<CUSTOMERTYPE>NT User</CUSTOMERTYPE>
<COMPANY>Microsoft</COMPANY> <VOICE>123 456 7890</VOICE> <ALTERNATIVE>789 123
4560</ALTERNATIVE>
<FAX>456 123 7890</FAX>
<COMMENTS>Very important customer</COMMENTS>
<PASSWORD>bill123</PASSWORD>
<AUTOPASSWD>False</AUTOPASSWD>
<ADDR1STR1>1234 John Fitzgerald</ADDR1STR1>
<ADDR1STR2>Kennedy Boulevard</ADDR1STR2>
<aDDR1CITY>Los Angeles</aDDR1CITY>
<ADDR1STATE>CA</ADDR1STATE> <ADDR1ZIP>12345</ADDR1ZIP>
<ADDR1COUNTRY>United States
```

```
<ADDR2STR1>456 Martin Luther</ADDR2STR1>
<ADDR2STR2>King Boulevard</ADDR2STR2>
<ADDR2CITY>San Francisco</ADDR2CITY>
<ADDR2 STATE>CA< /ADDR2 STATE> <ADDR2ZIP>67890</ADDR2ZIP>
<ADDR2COUNTRY>United States</ADDR2COUNTRY>
<ACCOUNTNUM>123456< /ACCOUNTNUM>
</ITEM>
<ITEM>
...
</ITEM>
</REPLY>
</WAPI>
```

Manipulating Customer Objects

You can use CRM API to add, modify, and get customer data. This section outlines the actions that can be performed on a customer object and the parameters that are used in both the request and the reply.

You can perform the following actions on the Customer object:

- Add
- Modify
- Get

Add Customer

Add Customer action allows a tenant to add a new customer in the CRM. This corresponds to the "Create" contact action in the Agent Console Interface.

- "Customer" keyword is not case sensitive.
- "Add" keyword is not case sensitive.

Example 7 illustrates an add customer request.

Example 7: An Add Customer request that includes custom fields

```
'<WAPI>
<tenant>AcmeJets</tenant>
<username>wapi</username>
<password>wapi1</password>
<COMMAND OBJECT="Customer" ACTION="Add">
<FIRSTNAME>Bill//FIRSTNAME>
<LASTNAME>Gates
<EMAIL>b.gates@microsoft.com</EMAIL>
<COMPANY>Microsoft</COMPANY>
<VOICE>123 456 7890</voice>
<ALTERNATIVE>789 123 4560</ALTERNATIVE>
<FAX>456 123 7890</FAX>
<COMMENTS>Very important customer</COMMENTS>
<AUTOPASSWD>FALSE</AUTOPASSWD>
<PASSWORD>bill123</PASSWORD>
<ADDR1STR1>1234 John Fitzgerald</ADDR1STR1>
<ADDR1STR2>Kennedy Boulevard</ADDR1STR2>
<ADDR1CITY>Los Angeles</ADDR1CITY>
<address <a href="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mailto:<ahref="mai
<ADDR1COUNTRY>United States</addr1COUNTRY>
<ADDR2STR1>456 Martin Luther/ADDR2STR1>
<ADDR2STR2>King Boulevard</ADDR2STR2>
<ADDR2CITY>San Francisco</ADDR2CITY>
<ADDR2STATE>CA</ADDR2STATE>
<addr2zip>67890</addr2zip>
<ADDR2COUNTRY>United States/ADDR2COUNTRY>
<CUSTOMERTYPE>Default
<CF01 PICKLIST.NAME>value1</CF01 PICKLIST.NAME>
<CF02.NAME>value2</CF02.NAME>
</COMMAND>
</WAPI>'
```

In the example results, CF01_PICKLIST. NAME and CF02.NAME identify previously defined custom Internal CRM fields.

Adding multiple customers or cases requires writing a program to enable the same. If you need to add multiple customers, modify the script to include multiple customer records

Mandatory and Optional Parameters

Table 3lists the mandatory and optional parameters.

- M Mandatory
- O Optional

Table 3: Summary of Add Customer parameters

Parameter	М	0	Format
FIRSTNAME	•		Character Max - 30
LASTNAME	•		Character Max - 30
EMAIL		•	(A-Z a-z - 0-9 _ @) Max - 55
			Note: Email address has to be unique for each customer instance.
COMPANY		•	Character Max - 50
			Invalid –(~,<,>, whitespace only)
VOICE		•	Valid- + () / 0-9 space. Max - 38
			Invalid – (whitespace only, on numeric characters)
ALTERNATIVE		•	Valid- + () / 0-9 space Max - 38
			Invalid – (whitespace only, non numeric characters)
FAX		•	Valid- + () / 0-9 space Max - 38'
			Invalid – (whitespace only, no numeric characters)
COMMENTS		•	Character Max - 1024
PASSWORD		•	A-Z, a-z,_,0-9 Min - 3 Max - 15
AUTOPASSWD	•		Possible values: "False" or "True" (case insensitive).
			If it is "False" the PASSWORD parameter must be provided. If it is "True" the PASSWORD parameter is ignored.
ADDR1 STR1		•	Any character Max – 50
ADDR1 STR2		•	Any character Max – 50
ADDR1CITY		•	Any character Max – 40
ADDR1 STATE		•	Any character Max - 30
ADDR1ZIP		•	Any character Max - 15
ADDR1COUNTRY		•	See the List of Attributes Values section to determine the list of countries.
			If no country is provided, United States is used.
ADDR2STR1		•	Any character Max - 50
ADDR2STR2		•	Any character Max - 50

Parameter	М	0	Format
ADDR2CITY		•	Any character Max - 40
ADDR2 STATE		•	Any character Max - 30
ADDR2ZIP		•	Any character Max – 15
ADDR2COUNTRY		•	See the List of Attributes Values section to determine the list of countries. If no country is provided, United States is used.
CUSTOMERTYPE		•	The contact type must be one valid type (not deactivated) as defined in the Configuration Manager. The list of possible values is defined by the Administrator of the tenant.
CF01_PICKLIST.NAME		•	Custom fields and their parameter names are added by the Administrator of the tenant. The values must be one valid type (not deactivated) as defined in the Configuration Manager. The list of possible values is defined by the Administrator of the tenant.
CF02.NAME		•	Custom fields and their parameter names are added by the Administrator of the tenant. The values must be one valid type (not deactivated) as defined in the Configuration Manager. The values passed should be in a text or number format.

Reply from CRM to Add Customer

If the customer record is added successfully, a response indicating the successful addition of a new customer along with the assigned Account Number is shown. Example 8 illustrates a reply to a request for adding a customer.

Example 8: Response to add customer operation

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<WAPI>
<REPLY ERROR_CODE="0" ERROR_STR="" STATUS="0">
<ITEM>
<ACCOUNTNUM>1453</ACCOUNTNUM>
</ITEM>
</REPLY>
</WAPI>
```

See Error Cases, on page 92 for a list of errors in an unsuccessful reply.

Modify Customer

Modify Customer modifies an existing Internal CRM customer record. This operation corresponds to the "edit" contact action in the Agent Supervisor Interface.

- Modify keyword is not case sensitive.
- Please note that the user may only send mandatory and modified parameters.
- Any parameter not supplied in a Modify request retains its previous value.
- Account Number is a mandatory field for modifying a customer record.

Example 9 illustrates a modify customer request. The request passes an updated email address.

Example 9: A modify customer operation that includes custom fields

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME> <PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Customer" ACTION="Modify">
<ACCOUNTNUM>1453</ACCOUNTNUM>
<FIRSTNAME>Bill</FIRSTNAME> <LASTNAME>Gates</LASTNAME>
<EMAIL>MSGuru@microsoft.com</EMAIL>
</COMMAND>
</WAPI>'
```

List of Mandatory and Optional Parameters.

Table 4lists the general form of a modify customer parameters.

- M Mandatory
- O Optional

Table 4 lists the general form of add customer parameters.

Table 4: Summary of Modify customer query values

Parameter	М	0	Format
ACCOUNTNUM	•		Customer account number to be modified (ACCOUNTNUM as returned by GET request).
FIRSTNAME		•	Character Max - 30
LASTNAME		•	Character Max - 30

Parameter	M	0	Format
EMAIL		•	(A-Z a-z – 0-9 _ @) Max - 55
COMPANY		•	Character Max - 50
			Invalid –(~,<,>, whitespace only)
VOICE		•	+ () / 0-9 space Max - 38
			Invalid – (whitespace only, no numeric characters)
ALTERNATIVE		•	+ () / 0-9 space Max - 38
			Invalid – (whitespace only, no numeric characters)
FAX		•	Valid- + () / 0-9 space
ACCOUNTNUM	•		Customer account number to be modified (ACCOUNTNUM as returned by GET request).
FIRSTNAME		•	Character Max - 30
LASTNAME		•	Character Max - 30
EMAIL		•	(A-Z a-z – 0-9 _ @) Max - 55
COMPANY		•	Character Max - 50
			Invalid –(~,<,>, whitespace only)
VOICE		•	+ () / 0-9 space Max - 38
			Invalid – (whitespace only, no numeric characters)
ALTERNATIVE		•	+ () / 0-9 space Max - 38
			Invalid – (whitespace only, no numeric characters)
FAX		•	Valid- + () / 0-9 space Max - 38
			Invalid – (whitespace only, no numeric characters)
COMMENTS		•	Character Max - 1024
PASSWORD		•	A-Z, a-z,_,0-9 Min - 3 Max - 15
AUTOPASSWD		•	Possible values: "False" or "True" (case insensitive).
			If it is "False" the PASSWORD parameter must be provided. If it is "True" the PASSWORD parameter is ignored.
ADDR1 STR1		•	Any character Max - 50
ADDR1 STR2		•	Any character Max - 50

Parameter	М	0	Format
ADDR1CITY		•	Any character Max - 40
ADDR1 STATE		•	Any character Max - 30
ADDR1ZIP		•	Any character Max - 15
ADDR1COUNTRY		•	See the List of Attributes Values section to determine the list of countries.
			If no country is provided, United States is used.
ADDR2STR1		•	Any character Max - 50
ADDR2STR2		•	Any character Max - 50
ADDR2CITY		•	Any character Max - 40
ADDR2 STATE		•	Any character Max - 30
ADDR2ZIP		•	Any character Max - 15
ADDR2COUNTRY		•	See the List of Attributes Values section to determine the list of countries.
			If no country is provided, United States is used
CUSTOMERTYPE		•	The contact type must be one valid type (not deactivated) as defined in the Configuration Manager. The list of possible values is defined by the Administrator of the tenant.
CF01_PICKLIST.NAME		•	Custom fields and their parameter names are added by the Administrator of the tenant. The values must be one valid type as defined in the Configuration Manager. The list of possible values is defined by the Administrator of the tenant.
CF02.NAME		•	Custom fields and their parameter names are added by the Administrator of the tenant. The values must be one valid type as defined in the Configuration Manager. The values passed should be in a text or number format.

Reply from CRM to Modify Customer

Example 10 illustrates a reply to of a customer modification request.

Example 10: Response to modify customer operation

<?xml version="1.0" encoding="ISO-8859-1"?>
<WAPI>

```
<REPLY ERROR_CODESTATUS="0" ERROR_STR="">
<ACCOUNTNUM>1453</ACCOUNTNUM>
</REPLY>
</WAPI>
```

See *Error Cases*, on page 92 for a list of errors to a unsuccessful reply.

Get Customer Details

This action allows a tenant to retrieve one or more customer details from the CRM. This corresponds to the "Search" and "View" links in the Agent Console Interface.

- **Get** keyword is not case sensitive.
- Command object only needs to be passed once, even when searching on multiple parameters.
- Parameter values being searched are case sensitive.
- A "GET" request can supply any combination of parameters to retrieve data.

Examples of Get Customer Operations

Example 11 illustrates a request to get customer details.

Example 11: Customer detail search for company name

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME> <PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Customer" ACTION="GET">
<COMPANY>microsoft</COMPANY>
</COMMAND>
</WAPI>'
```

To retrieve customer records where company name includes a particular letter, use the percent sign % to specify a single wild card character.

Example 12 illustrates a customer details search that uses a single wild card character to return all customer records where COMPANY begins with m.

Example 12: Customer detail search with wild card for company names containing m

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Customer" ACTION="GET"> <COMPANY>m%</COMPANY>
</COMMAND>
</WAPI>'
```

Example 13 illustrates a customer details search for a particular last name.

Example 13: Customer detail search for a last name

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Customer" ACTION="GET">
<LASTNAME>gates</LASTNAME>
</COMMAND>
</WAPI>'
```

Example 14 illustrates a customer details search that uses a single wild card character to return all customer records where LASTNAME beginning with g.

Example 14: Search with Wild card for last names that contain g

```
'<WAPI>

<TENANT>AcmeJets</TENANT>

<USERNAME>johnd</USERNAME>

<PASSWORD>john123</PASSWORD>

<COMMAND OBJECT="Customer" ACTION="GET"> <LASTNAME>g%</LASTNAME>

</COMMAND>

</WAPI>'
```

Example 15 illustrates a customer details search for a specific email address.

Example 15: Customer detail search for an email address

```
'<WAPI>
<TENANT>C ompanyX</TENANT> <USERNAME>johnd</USERNAME> <PASSWORD>john12 3</PASSWORD>
<COMMAND OBJECT="Customer" ACTION="GET"> <EMAIL>bill@microsoft.com</EMAIL>
</COMMAND>
</WAPI>'
```

Example 16 illustrates a customer details search that uses a single wild card character to return all customer records where EMAIL begins with b.

Example 16: Customer detail search with wild card for email addresses that contain b

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Customer" ACTION="GET"> <EMAIL>b%</EMAIL>
</COMMAND>
</WAPI>'
```

Example 17 illustrates a customer details search for a specific contact type. The contact type must be one valid, not deactivated type defined in the Configuration Manager.

Example 17: Customer detail search for customer type

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Customer" ACTION="GET">
<CUSTOMERTYPE>NT User</CUSTOMERTYPE>
</COMMAND>
</WAPI>'
```

Example 18 illustrates a customer details search for a specific account number.

Example 18: Customer detail search for a specifc account number

```
'<WAPI>
<TENANT>AcmeJets</free/TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Customer" ACTION="GET"> <ACCOUNTNUM>1453</ACCOUNTNUM>
</COMMAND>
</WAPI>'
```

Example 19 illustrates a customer detail search that uses a wild card construct to return all customer records that contain 45.

Example 19:

```
'<WAPI>
<TENANT>AcmeJets</free/TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Customer" ACTION="GET"> <ACCOUNTNUM>%45%</ACCOUNTNUM>
</COMMAND>
</WAPI>'
```

A generic search of customer records returns:

- A range of available account numbers
- The total number of available records

Example 20 illustrates the general form of the XML used to perform customer record searches.

Example 20:

```
...
<FIRSTID>ID of first record available</FIRSTID>
<LASTID>ID of last record available</LASTID>
<TOTAL>total number of records</TOTAL>
...
```

Example 21 illustrates a search for all customer records in a tenant. The response includes information on the Account Number range of available records as well as the total number of customer records.

Example 21:

```
'<WAPI>
TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Customer" ACTION="GET">
</COMMAND>
</WAPI>'
```

Example 22 illustrates the Internal CRM response to the customer detail search.

Example 22:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<WAPI>
<REPLY ERROR_CODE="0" ERROR_STR="" STATUS="0" >
<ITEM>
<FIRSTID>10000000</FIRSTID>
<LASTID>10000083</LASTID>
<TOTAL>81</TOTAL>
</ITEM>
</REPLY>
</WAPI>
```

See *Error Cases*, on page 92 for a list of errors to a unsuccessful reply.

In the Internal CRM, the contact center administrator uses the Configuration Manager to define custom fields and parameters.

Example 23 illustrates a customer details search for the custom picklist name CF01.

Example 23:

```
'<WAPI>
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Customer" ACTION="GET">
<CF01_PICKLIST.NAME>Value</CF01_PICKLIST.NAME>
</COMMAND>
</WAPI>'
```

Example 24 illustrates a customer details search for custom field CF02 that contains a specific Value.

Example 24: Customer detail search of custom field containing specific value

```
'<WAPI>

<TENANT>AcmeJets</TENANT> <USERNAME>johnd</USERNAME> <PASSWORD>john123</PASSWORD>

<COMMAND OBJECT="Customer" ACTION="GET">

<CF02.NAME>Value</CF02.NAME>

</COMMAND>

</WAPI>'
```

List of Parameters

Table 5illustrates the customer details parameters.

At least one of the parameters must be used in a Customer Details GET request.

Table 5: List of customer detail parameters

Parameter	Format			
FIRSTNAME	Character Max - 30			

Parameter	Format				
LASTNAME	Character Max - 30				
EMAIL	(A-Z a-z - 0-9 _ @) Max - 55				
VOICE	+ () / 0-9 space				
	Max - 38				
	Invalid – (whitespace only, no numeric characters)				
COMPANY	Character				
	Max - 50				
	Invalid – (~,<,>, whitespace only)				
ACCOUNTNUM	Account number of the customer who is related to the case. This account number is returned after a "GET" action on a "Customer" object.				
CUSTOMERTYPE	The contact type must be one valid type (not deactivated) as defined in the Configuration Manager. The list of possible values is defined by the Administrator of the tenant.				
CF01_PICKLIST.NAME	Custom fields and their parameter names are added by the Administrator of the tenant.				
	The values must be one valid type (not deactivated) as defined in the Configuration Manager. The list of possible values is defined by the Administrator of the tenant.				
CF02.NAME	Custom fields and their parameter names are added by the Administrator of the tenant.				
	The values must be one valid type (not deactivated) as defined in the Configuration Manager. The list of possible values is defined by the Administrator of the tenant.				

See Error Cases, on page 92 for a list of errors to a unsuccessful reply.

Reply from CRM to Get Customer Details

Example 25 illustrates a reply to a get customer details request.

In Example 25, CF01_PICKLIST . NAME and CF02 . NAME are custom fields.

Example 25: Response to get customer details operation

```
<?xml version="1. 0" encoding="ISO-8859-1"?>
<WAPI>
<REPLY STATUS=0 ERROR STR=""> <ITEM>
<FIRSTNAME>Bi 1 1</firstname> <LASTNAME>Gates
<EMAIL>bill . gates@microsoft. com</EMAIL>
<COMPANY>Micro soft</COMPANY> <VOICE>123 456 7890</VOICE> <ALTERNATIVE>789 123
4560</ALTERNATIVE>
<FAX>456 123 7890</FAX>
<COMMENTS>Very important customer</COMMENTS> <PASSWORD>bill123</PASSWORD>
<ADDR1STR1>1234 John Fitzgerald</ADDR1STR1> <ADDR1STR2>Kennedy
Boulevard</ADDR1STR2>
<address{ADDR1CITY>Los Angeles</addressTY>
<ADDR1 STATE>CA</ADDR1 STATE> <ADDR1ZIP>12345</ADDR1ZIP> <ADDR1COUNTRY>United
States</ADDR1COUNTRY>
<ADDR2STR1>4 56 Martin Luther</aDDR2STR1>
<ADDR2STR2>King Boulevard</ADDR2STR2>
<ADDR2CITY>San Francisco</ADDR2CITY>
<ADDR2 STATE>CA< /ADDR2 STATE> <ADDR2ZIP>678 90</ADDR2ZIP> <ADDR2COUNTRY>United
States</ADDR2COUNTRY>
<ACCOUNTNUM> 123456< /ACCOUNTNUM> <CUSTOMERTYPE>NT user< /CUSTOMERTYPE>
<CF01 PICKLIST .NAME>value1< /CF01 PICKLIST .NAME>
<CF02 .NAME>value2</CF02 .NAME> ...
</ITEM>
<ITEM>
</ITEM>
</REPLY>
</WAPI>
```

See *Error Cases*, on page 92 for a list of errors to a unsuccessful reply.

Manipulating the Case Object

You can use CRM API to add, modify, and get case data. The section outlines all the possible actions that can be performed on a Case object and all the parameters that are used in both the request and the reply.

You can perform the following actions on the Case object:

- Add
- Modify
- Get

Add Case

This action enables a tenant to add a new case to the CRM. This action corresponds to the "create" case action in the Agent Console Interface.

- "Case" keyword is not case sensitive.
- "Add" keyword is not case sensitive.
- Account Number, Subject and Description are mandatory fields to add a case.

Example 26 illustrates an add case operation.

Note: In Example 26, CF01 PICKLIST NAME and CF02.NAME are custom fields.

Example 26: Add case operation

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</username>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="Add">
<accountnum>10000019</accountnum>
<SUBJECT>My printer doesn't work</SUBJECT>
<DESCRIPTION>This customer downloaded the new driver
XXXX v5.4 of the driver, installed it and since then the
printer doesn't work.</DESCRIPTION>
<STATUS>Unassigned</STATUS>
<PRIORITY>Medium</PRIORITY>
<SEVERITY>Information
<PROJECT>default</PROJECT>
<CATEGORY>Default</CATEGORY>
<VISIBILITY>Internal</VISIBILITY>
<MEDIATYPE>Phone
<CF01 PICKLIST.NAME>value1</CF01 PICKLIST.NAME>
<CF02.NAME>value2</CF02.NAME>
</COMMAND>
</WAPI>'
```

List of Mandatory and Optional Parameters

Table 6 summarizes the add case mandatory and optional parameters.

- M Mandatory
- O Optional

Table 6: Summary of add case parameters

Parameter	М	0	Format
ACCOUNTNUM	•		Account number of the customer who is related to the case. This account number is returned after a "GET" action on a "Customer" object.
SUBJECT	•		Any character
			Max - 255
DESCRIPTION			Invalid – whitespace only
DESCRIPTION	•		■ Any character
			Max - 2mb
			■ Invalid – whitespace only.
STATUS		•	The status must be one valid status (not deactivated) as defined in the Configuration Manager.
			The "GET" action on a Custom Field object can be used to determine the list of possible statuses.
			If no status is provided, the default value is "unassigned". If status is used, the "assignedto" parameter must also be defined.
PRIORITY		•	The priority must be one valid priority (not deactivated) as defined in the Configuration Manager. The list of possible values is defined by the Administrator of the tenant.
			The "GET" action on a Custom Field object can be used to determine the list of possible priorities.
			If no priority is provided, the default defined one is used.
SEVERITY		•	The severity must be one valid severity (not deactivated) as defined in the Configuration Manager. The list of possible values is defined by the Administrator of the tenant.
			The "GET" action on a Custom Field object can be used to determine the list of possible severities.
			If no severity is provided, the default defined one is used.
PROJECT		•	The project must be a valid project (not deactivated) as defined in the Configuration Manager. The list of possible values is defined by the Administrator of the tenant.
			The "GET" action on a Custom Field object can be used to determine the list of possible projects.

Parameter	М	О	Format
			If no project is provided, the default defined one is used.
CATEGORY		•	The category must be one valid category (not deactivated) as defined in the Configuration Manager. The list of possible values is defined by the Administrator of the tenant. The "GET" action on a Custom Field object can be used to determine the list of possible categories.
			If no category is provided, the default defined one is used.
VISIBILITY		•	The visibility must be one valid category (not deactivated) as defined in the Configuration Manager. Possible values are "Internal" or "External". If no visibility is
			provided, the "Internal" value is used.
MEDIATYPE		•	The mediatype must be one valid media type (not deactivated) as defined in the Configuration Manager.
			Possible values are: "Phone", "Chat", "Email", "Support Center" or "None".
ASSIGNEDTO		•	The "assignedto" must be equal to a valid agents or groups user id. If "assignedto" is used, the "status" parameter must also be defined.
			To search on cases assigned to deleted agents, deleted groups, or unassigned agents the values are:
			"Deleted Agents", "Deleted Groups", or "None".
CF01_PICKLIST.NAME		•	Custom fields and their parameter names are added by the Administrator of the tenant. The values must be one valid type (not deactivated) as defined in the Configuration Manager. The list of possible values is defined by the Administrator of the tenant.
CF02.NAME		•	Custom fields and their parameter names are added by the Administrator of the tenant. The values must be one valid type (not deactivated) as defined in the Configuration Manager. The values passed should be in a text or number format.

See Error Cases, on page 92 for a list of errors to a unsuccessful reply.

Reply from CRM to Add Case

Example 27 illustrates a reply to add a new case.

Example 27: Response to add case operation.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<WAPI>
<REPLY STATUS="0" ERROR _STR="">
<CASENUM>67890</CASENUM>
</REPLY>
</WAPI>
```

See Error Cases, on page 92 for a list of errors to a unsuccessful reply.

Modify Case

The Modify Case operation modifies an existing Internal CRM case record. This corresponds to the "edit" case action in the Agent ConsoleInterface.

- "Modify" keyword is not case sensitive.
- The user may only send mandatory and modified parameters.
- Any parameter not supplied in a "Modify" request will retain the previous value.
- Any optional parameter not supplied in a "Modify" request will be set to null in the CRM.
- Case Number is mandatory to modify a case.

Case Modification generates an automated follow-up containing a description of the modification consistent with an edit case action taken through the Agent Console Interface.

Example 28 illustrates a modify case request.

In the example, CF01_PICKLIST . NAME and CF02 . NAME identify previously identified custom Internal CRM fields.

Example 28: A modify case operation that includes custom fields

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="Modify">
<CASENUM>67890</CASENUM>
<STATUS>Open</STATUS>
<PRIORITY>High</PRIORITY>
<ASSIGNEDTO>jsmith</ASSIGNEDTO>
</COMMAND>
<WAPI>'
```

List of Mandatory and Optional Parameters

Table 7 summarizes the mandatory and optional modify case parameters

- M Mandatory
- O Optional

Table 7: Summary of optional and mandatory modify case parameters

Parameter	М	0	Format
CASENUM	•		This case number is returned after a "GET" action on a "Case" object.
SUBJECT		•	Any character
			Max - 255
			Invalid – whitespace only
DESCRIPTION		•	Any character
			Max - 2mb
			Invalid – whitespace only
STATUS		•	The status must be one valid status (not deactivated) as defined in the Configuration Manager.
			The "GET" action on a Custom Field object can be used to determine the list of possible statuses.
			If no status is provided, the default value is "unassigned". If status is assigned, the "assignedto" parameter must also be defined.
PRIORITY		•	The priority must be one valid priority (not deactivated) as defined in the Configuration Manager. The list of possible values is defined by the Administrator of the tenant.
			The "GET" action on a Custom Field object can be used to determine the list of possible priorities.
			If no priority is provided, the default defined one is used.
SEVERITY		•	The severity must be one valid severity (not deactivated) as defined in the Configuration Manager. The list of possible values is defined by the Administrator of the tenant.
			The "GET" action on a Custom Field object can be used to determine the list of possible severities.
			If no severity is provided, the default defined one is used.
CATEGORY		•	The category must be one valid category (not deactivated) as defined in the Configuration Manager. The list of possible values is defined by the Administrator of the tenant.
			The "GET" action on a Custom Field object can be used to determine the list of possible categories.
			If no category is provided, the default defined one is used.

Parameter	М	O	Format
VISIBILITY		•	The visibility must be one valid category (not deactivated) as defined in the Configuration Manager. Possible values are Internal or External. If no visibility is provided, the "Internal" value is used.
MEDIATYPE		•	The mediatype must be one valid media type (not deactivated) as defined in the Configuration Manager. Possible values are: <i>Phone, Chat, Email, Support Center</i> or <i>None</i> .
ASSIGNEDTO		•	The "assignedto" must be equal to an agents or group user id. If "assignedto" is used, the "status" parameter must also be defined.

See Error Cases, on page 92 for a list of errors to a unsuccessful reply.

Reply from CRM to Modify Case

Example 29 illustrates a reply to a modify existing case request.

Example 29: Response to modify customer operation

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<WAPI>
<REPLY ERROR_CODE="0" ERROR _STR="" STATUS="0">
<CASENUM>67890</CASENUM>
</REPLY>
</WAPI>
```

See Error Cases, on page 92 for a list of errors to a unsuccessful reply.

Get Case

This action enables a tenant to get a list of existing cases from the CRM that matches the supplied parameter values. This action corresponds to the "Search", "View" and "List" actions in the Agent Console Interface.

- **Get** keyword is not case sensitive.
- Command object only needs to be passed once, even when searching on multiple parameters.
- Parameter values being searched are case sensitive.

■ This request returns entire descriptions for each case that matches the supplied parameter values, including the customer account number.

The query parameters on a "GET" request can vary in the following ways:

- Query on a substring
- Pass multiple parameters to generate an "OR" query
- Pass multiple parameters to generate a "range" query

Tags that accept substring values

- Last Name
- Subject
- Company

Parameters values being searched as a substring require the use of %. For example, to search for the word printer, the search string %printer%.

Tags that accept multiple parameters

- CreatedBy
- AssignedTo
- Status
- Category
- Priority
- Severity
- Project

Tags that can be paired for "range" queries and creation dates

- FromDate
- ToDate

To retrieve cases created on a particular date, "FromDate" should correspond to the first date desired and the "ToDate" should correspond to the day after.

For example, a "FromDate" of '12072001' to a "ToDate" of '12082001' will retrieve cases created December 7th, 2001.

Examples of Get Case Operations

Example 30 illustrates a get case search given a case number.

Example 30: Get case search by case number

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET"> <CASENUM>67890</CASENUM>
</COMMAND>
</WAPI>'
```

Example 31 illustrates a get case search for case STATUS equal to Unassigned.

The case status must match the values defined in the Configuration Manager. The "GET" action on a Custom Field object can be used to determine the list of possible statuses.

Example 31: Get case search by case status

```
'<WAPI>
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET"> <STATUS>Unassigned</STATUS>
<STATUS>Open</STATUS>
</COMMAND>
</WAPI>'
```

Example 32 illustrates a get case search by creation date.

Only one occurrence of the FROMDATE and TODATE parameters can be used in one GET request.

To retrieve cases created on a particular date, "FromDate" should correspond to the first date desired and the "ToDate" should correspond to the day after. For example, a "FromDate" of '12072001' to a "ToDate" of '12082001' retrieves cases created on December 7th, 2001.

Example 32: Get case search by creation date

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET">
<FROMDATE>01112010</FROMDATE>
<TODATE>02112001</TODATE>
</COMMAND>
</WAPI>'
```

Example 33 illustrates a case search by category.

You can retrieve cases that match multiple category types. The categories must be active and match the values defined in the Configuration Manager. The "GET" action on a Custom Field object can be used to determine the list of possible categories.

Example 33: Get case search by case category

```
'<WAPI>
'<WAPI>
<TENANT>AcmeJets</TENANT> <COMMAND OBJECT="Case" ACTION="GET">
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET">
<CATEGORY>Gold</CATEGORY> <CATEGORY>Platinum</CATEGORY>
</COMMAND>
</WAPI>'
```

Example 34 fetches cases assigned to specific agents.

Example 34: Get case search by assigned agent

```
'<WAPI>
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET">
<ASSIGNEDTO>jsmith</ASSIGNEDTO>
<ASSIGNEDTO>rshasta</ASSIGNEDTO>
<ASSIGNEDTO>sbaker</ASSIGNEDTO>
</COMMAND>
</WAPI>'
```

Example 35 fetches cases created by specific agents.

Example 35: Get case search by creator

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET">
<CREATEDBY>jsmith</CREATEDBY>
<CREATEDBY>jdoe</CREATEDBY>
</COMMAND>
</WAPI>'
```

To retrieve cases where the subject includes a particular character string, enclose the string with % signs.

Only one occurrence of this tag can be used in one GET request. Parameters values being searched as a substring require the use of "%". For example, to search on the word "printer", the value should be set as: "%printer%".

Example 36 illustrates a search for cases whose SUBJECT line contains the substring printer.

Example 36: Get case search by subject using wildcard characters to search by substring

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET">
<SUBJECT>%printer%</SUBJECT>
</COMMAND>
</WAPI>'
```

Example 37 illustrates a search for cases using a wild card character to find cases with SUBJECT beginning with p.

Example 37: Get case search by subject using wildcard characters

```
'<WAPI>
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET">
<SUBJECT>p %</SUBJECT>
</COMMAND>
</WAPI>'
```

Example 38 illustrates a search for cases attributed to a company. The search using wild card characters returns all records where COMPANY contains the substring Microsoft.

Only one occurrence of this tag can be used in one GET request. Parameters values being searched as a substring require the use of "%". For example, to search on the word "Microsoft", the value should be set as: "%Microsoft".

Example 38: Get case search by company using wildcard characters to search by substring

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET">
<COMPANY>%Microsoft%</COMPANY>
</COMMAND>
</WAPI>'
```

Example 39 illustrates a get case search using wild card characters to return all records where COMPANY begins with m.

Only one occurrence of this tag can be used in one GET request. Parameters values being searched as a substring require the use of "%". For example, to search for a portion of a parameter value such as "Microsoft" in "MicroSoft Corporation", the value should be set as: "%Microsoft%". You should not to use the wild card search if you provide the whole parameter value.

Example 39: Get case search by company using wildcard characters

```
'<wapi>
'<wapi>
<TENANT>AcmeJets</TENANT>

<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>

<COMMAND OBJECT="Case" ACTION="GET">

<COMPANY>%m%</COMPANY>
</COMMAND>
</WAPI>'
```

Example 40 illustrates a search for cases using wildcard characters where LASTNAME contains the substring gates.

Example 40: Get case search by last name using wild card characters to search by substring

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET"> <LASTNAME>%gate%</LASTNAME>
</COMMAND>
</WAPI>'
```

Example 39 illustrates a get case search using wild card characters that fetches all records where LASTNAME begins with g.

Example 41: Get case search by last name using wild card characters

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET"> <LASTNAME>g%</LASTNAME>
</COMMAND>
</WAPI>'
```

To fetch information on the total number of cases in a tenant, you can send a GET request without specifying any parameters. Example 42 illustrates a search that fetches a brief summary of cases in the tenant.

Example 42: Get case search without specifying parameters

```
'<WAPI>

'TENANT>AcmeJets</free/TENANT>

<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET">
</COMMAND>
</WAPI>'
```

The search yields a brief summary of cases in the tenant including the Case ID of first case, Case ID of the last case, and the total number of cases.

Example 43 illustrates the general form of the information returned by the CRM API when searching a case without providing ANY parameters.

Example 43: General form of results of get case search without qualifying parameters

```
<FIRSTID>ID of first case record available</FIRSTID>
<LASTID>ID of last case record available</LASTID>
<TOTAL>total number of case records</TOTAL>
```

Example 44 illustrates the results of a case search performed without any parameters. In Example 44, the first case records equals 1, the last case record equals 100, and the total number of active cases equals 90.

Example 44: Results of get case search without any parameters

```
<?xml version="1 . 0" encoding="ISO-8859-1"?>
<WAPI>
<REPLY STATUS="0" ERROR _STR="">
<ITEM>
<ITEM>
<FIRST ID>1001</FIRSTID> <LASTID>70030</LASTID>
<TOTAL>90</TOTAL>
</ITEM>
</REPLY>
</WAPI>
```

Example 45 illustrates a search for cases that match a value specified by a custom field of Picklist datatype. In the Internal CRM, the contact center administrator uses the Configuration Manager to define custom fields and parameters.

Example 45: Get case search by a custom field containing a picklist value

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET">
<CF01_PICKLIST.NAME>Value</CF01_PICKLIST.NAME>
</COMMAND>
</WAPI>'
```

Example 46 illustrates a get case search for cases created by Customers.

"Customers" is the value the CRM API user has to specify to fetch all the cases created by customers. This value is not case sensitive.

Example 46: Get case search by Customers

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET"> <CREATEDBY>Customers</CREATEDBY>
</COMMAND>
</WAPI>'
```

To retrieve cases assigned to a Group, you must get the group ID assigned to the group by the CRM. To access the group ID of a given group, run a request to list the groups. See "List Agent GROUPS" on page 88.

Example 47 illustrates a search for cases assigned to Sales group. The Group ID for sales returned by the CRM is 101. Group names are defined by the Administrator of the tenant in the Configuration Manager.

Example 47: Get case search by cases assigned to a group

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET"> <ASSIGNEDTO>101</ASSIGNEDTO>
</COMMAND>
</WAPI>'
```

Example 48 illustrates a get case search for cases assigned to deleted groups.

"Deleted Groups" is the value the CRM API user has to specify to fetch all the cases assigned to deleted Groups. This value is not case sensitive. Only one space character should separate "Deleted" and "Groups".

Example 48: Get case search by cases assigned to deleted groups

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET">
<ASSIGNEDTO>Deleted Groups</ASSIGNEDTO>
</COMMAND>
</WAPI>'
```

Example 49 illustrates a get case search for cases created by deleted agents.

"Deleted Agents" is the value the CRM API user has to specify to fetch cases created by deleted agents. This value is not case sensitive. Only one space character should separate "Deleted" and "Agents". This is useful since your search for all cases does not automatically yield cases created by deleted agents.

Example 49: Get case search for cases created by deleted agents

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET">
<CREATEDBY>Deleted Agents</CREATEDBY>
</COMMAND>
</WAPI>'
```

Example 50 illustrates a get case search for cases assigned to deleted agents.

Example 50: Get case search by cases assigned to deleted agents

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET">
<ASSIGNEDTO>Deleted Agents</ASSIGNEDTO>
</COMMAND>
</WAPI>'
```

Example 51 illustrates a get case search for unassigned cases.

Example 51: Get case search by unassigned cases

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET"> <ASSIGNEDTO>None</ASSIGNEDTO>
</COMMAND>
</WAPI>'
```

List of Parameters

Table 8 summarizes the get case parameters.

At least one parameter must be used in a GET request.

Table 8: List of get case parameters

Parameter	Format				
CASENUM	This case number is returned after a "GET" action on a "Case" object.				
STATUS	The status must be one valid status (not deactivated) as defined in the Configuration Manager.				
	The "GET" action on a Custom Field object can be used to determine the list of possible statuses.				
FROMDATE	The format for this attribute is:				
	MMDDYYYY where MM: Month				
	DD: Day				
	YYYY: Year.				

Parameter	Format
TODATE	The format for this attribute is:
	MMDDYYYY where MM: Month
	DD: Day
	YYYY: Year.
CATEGORY	The category must be one valid category (not deactivated) as defined in the Configuration Manager. The list of possible values is defined by the Administrator of the tenant.
	The "GET" action on a Custom Field object can be used to determine the list of possible categories.
PRIORITY	The priority must be one valid priority (not deactivated) as defined in the Configuration Manager. The list of possible values is defined by the Administrator of the tenant.
	The "GET" action on a Custom Field object can be used to determine the list of possible priorities.
SEVERITY	The severity must be one valid severity (not deactivated) as defined in the Configuration Manager.
	The list of possible values is defined by the Administrator of the tenant.
	The "GET" action on a Custom Field object can be used to determine the list of possible severities.
PROJECT	The project must be a valid project (not deactivated) as defined in the Configuration Manager. The list of possible values is defined by the Administrator of the tenant.
	The "GET" action on a Custom Field object can be used to determine the list of possible categories.
ASSIGNEDTO	The assigned to must be equal to a valid agents or groups user id.
	To search on cases assigned to deleted agents, deleted groups, or unassigned agents the values are:
	"Deleted Agents", "Deleted Groups", or "None".
CREATEDBY	The agent must be a valid agent or group user id. See the Listing Tenant Configuration Attributes section to get a list of agents and groups.
	To search a list of all customers, the generic value "Customers" can be used.
	To search cases created by deleted agents, the generic value "Deleted Agents" can be used.

Parameter	Format					
	Note: When cases are created by customers, the "CREA TEDBY" tag will be returned empty.					
SUBJECT	Any character					
	Max - 255					
	Invalid – Whitespace only					
COMPANY	Character					
	Max - 50					
	Invalid – (~ ,<>,whitespace only)					
LASTNAME	Character Max - 30 Invalid – (~ ,<>,whitespace only)					
ATTACHMENTS	Possible values: "yes" or "no"					
CFO _ PICKLIST.NAME	Custom fields and their parameter names are added by the Administrator of the tenant.					
	The values must be one valid type (not deactivated) as defined in the Configuration Manager. The list of possible values is defined by the Administrator of the tenant.					
CFO2.NAME	Custom fields and their parameter names are added by the Administrator of the tenant.					
	The values must be one valid type (not deactivated) as defined in the Configuration Manager. The values passed should be in a text or number format.					

Reply from CRM to Get Case

Example 52 illustrates a to a request for an existing case.

In Example 52, CF01_PICKLIST . NAME and CF02 . NAME are custom fields.

Example 52: Reply to successful Get Case Operation

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<WAPI>
<REPLY ERROR_CODE="0" ERROR_STR="" STATUS="0">
<ITEM>
<accountnum>123456</accountnum>
<CASENUM>67890</CASENUM>
<SUBJECT>My printer doesn't work</SUBJECT>
<DESCRIPTION>This customer downloaded the new driver XXXX v5.4 of the driver,
installed it and since then the printer doesn't work.</DESCRIPTION>
<STATUS>Closed</STATUS>
<PRIORI TY>High</PRIORITY>
<SEVERITY>Information
<CATEGORY>Default</CATEGORY>
<PROJECT>Default</PROJECT>
<VISIBILITY>Private</VISIBILITY>
<MEDIATYPE>Phone</MEDIATYPE>
<ASSIGNEDTO>jsmith</ASSIGNEDTO>
<ASSIGNEDDATE>01122010</ASSIGNEDDATE>
<CREATEDBY>bpower</CREATEDBY>
<CREATEDDATE>0 215200 0</CREATEDDATE>
<CLOSEDBY>bpower</CLOSEDBY>
<CLOSEDDATE>01232000< /CLOSEDDATE>
<LASTACTDATE>01232000
<CF01 PICKLIST.NAME>value1< /CF01 PICKLIST.NAME>
<CF02.NAME>value2</CF02.NAME>
</ITEM>
<ITEM>
</ITEM>
</REPLY>
</WAPI>
```

See *Error Cases*, on page 92 for a list of errors to a unsuccessful reply.

Manipulating the Follow-up Object

The section describes the actions that can be performed on a Internal CRM Follow-up object, including the parameters used in both the request and the reply.

For Follow-up objects, the following actions can be performed:

- Add
- Get

Add Followup

This action allows a tenant to add a followup to a case in the CRM. This corresponds to the "Create" followup action in the Agent Console Interface.

- "Followup" keyword is not case sensitive.
- "Add" keyword is not case sensitive.

Example 53 illustrates an add followup.

In Example 53, CF01 PICKLIST . NAME and CF02 . NAME are custom fields.

Example 53: An Add Followup request

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>wapi</USERNAME>
<PASSWORD>wapi1</PASSWORD>
<COMMAND OBJECT="Followup" ACTION="Add">
<CASENUM>35</CASENUM>
<SUBJECT>New Order placed</SUBJECT>
<Description>This customer downloaded the new driver XXXX v5.4 of the driver,
installed it and since then the printer doesn't work.</Description>
<STATUS>Open</STATUS>
<PRIORITY>Medium</PRIORITY>
<SEVERITY>Information</severity>
<Project>Default</Project>
<CATEGORY>Default</CATEGORY>
<VISIBILITY>Internal</VISIBILITY>
<MEDIATYPE>Voice</MEDIATYPE>
<ASSIGNEDTO>jsmith</ASSIGNEDTO>
</COMMAND>
</WAPI>'
```

List of Mandatory and Optional Parameters

Table 9 summarizes the add case mandatory and optional parameters.

- M Mandatory
- O Optional

Table 9: Summary of add followup parameters

Parameter	М	o	Format
CASENUM	•		Case number related to the follow-up. This case number is returned after a "GET" action on a Case object.
SUBJECT	•		Any character
			Max - 255
			Invalid – White space only
DESCRIPTION	•		Any character
			Max - 2mb
			Invalid – White space only
STATUS		•	The status must be a valid value (not deleted) as defined in the Configuration Manager.
			The "GET" action on a Custom Field object can be used to determine the list of possible statuses.
PRIORITY		•	The priority must bea valid value(not deleted) as defined in the Configuration Manager.
			The list of possible values is defined by the Administrator of the tenant.
			The "GET" action on a Custom Field object can be used to determine the list of possible priorities.
SEVERITY		•	The severity must be a valid value (not deleted) as defined in the Configuration Manager.
			The list of possible values is defined by the Administrator of the tenant.
			The "GET" action on a Custom Field object can be used to determine the list of possible severities.
PROJECT		•	The project must be a valid project (not deleted) as defined in the Configuration Manager.
			The list of possible values is defined by the Administrator of the tenant.
			The "GET" action on a Custom Field object can be used to determine the list of possible projects.
CATEGORY		•	The category must be one valid category (not deleted) as defined in the Configuration Manager.
			The list of possible values is defined by the Administrator of the

Parameter	М	О	Format
			tenant.
			The "GET" action on a Custom Field object can be used to determine the list of possible categories.
VISIBILITY		•	The visibility must be one valid value (not deleted) as defined in the Configuration Manager.
			Possible values are "Internal" or "External".
			If no visibility is provided, the default value is used.
MEDIATYPE		•	The mediatype must be one valid media type (not deleted) as defined in the Configuration Manager.
			Possible values are: "Phone", "Chat", "Email", "Support Center" or "Other".
			If no mediatype is provided, the default value is used.
ASSIGNEDTO		•	The "assignedto" must be equal to a valid agents or groups user id.
			If "assignedto" is used, the "status" parameter should be consistent.
			To search on cases assigned to deleted agents deleted groups, or unassigned cases the values are: "Deleted Agents", "Deleted Groups", or "None".
CF01_PICKLIST.NAME		•	Custom fields and their parameter names are added by the Administrator of the tenant.he values must be valid(not deactivated) and match the values defined in the Configuration Manager. The list of possible values is defined by the Administrator of the tenant.
CF02.NAME		•	Custom fields and their parameter names are added by the Administrator of the tenant. The values must be valid(not deactivated) and match the values defined in the Configuration Manager. The values passed should be in a text or number format.

Case Modification generates automated follow-ups containing a description of the modification consistent with an edit action taken through the Agent Console Interface.

Reply from CRM to Add Followup

Example 54 illustrates a reply to a request to add a new followup.

Example 54: Response to add followup

```
<?xml version="1. 0" encoding="ISO-8859-1"?>
<WAPI>
REPLY ERROR_CODE="0" ERROR_STR="" STATUS="0">
<FOLLOWUPNUM>11</FOLLOWUPNUM>
</REPLY>
</WAPI>
```

See *Error Cases*, on page 92 for a list of errors to a unsuccessful reply.

Get Followup

This action enables a tenant to get a list of existing followups in the CRM that matches the supplied parameter values. This action corresponds to the "Search", "View" and "List" actions in the Agent Console Interface.

- Get keyword is not case sensitive.
- Command object only needs to be passed once, even when searching on multiple parameters.
- Case Number is a mandatory field to fetch follow-up data.
- Follow-up records cannot be referenced by any field other than the case number.
- When searching followups, only one case number can be provided in the GET request.
- Wildcard search is limited to case numbers.

Example 55 illustrates searching followups for a specific case.

Example 55: Get followup search by case number

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Followup" ACTION="GET">
<CASENUM>67890</CASENUM>
</COMMAND>
</WAPI>'
```

Example 56 illustrates a wild card search for followups to all CASENUM containing the substring 78.

Example 56: Get followup search using wild card characters to search by substring

```
<WAPI>
<TENANT>AcmeJets</fr>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Followup" ACTION="GET">
<CASENUM>*78*</CASENUM>
</COMMAND>
</WAPI>
```

List of Parameters

The CRM API allows referencing follow-up records only by case numbers.

Table 10 summarizes the get followup parameter.

Table 10: Summary of get followup parameter

Parameter	Format
CASENUM	Case number related to the follow-ups. This case number is returned after a "GET" action on a "Case" object.

Reply from CRM to Get Followup

Example 57 illustrates a reply to a request for an existing followup.

In Example 57, CF01_PICKLIST . NAME and CF02 . NAME are custom fields.

Example 57: Response to Get Followup operation

```
<?xml version="1.0" encoding="ISO-8859-1"?>
-<WAPI>
-<REPLY ERROR CODE="0" ERROR STR="" STATUS="0">
-<ITEM><CASENUM>35</CASENUM>
<FOLLOWUPNUM>3</FOLLOWUPNUM>
<SUBJECT>Order Received - Issue Resolved</SUBJECT>
<CREATEDDATE>0503201212:26:59</CREATEDDATE>
<artachments>0</artachments>
<FOLLOWUPID>5</FOLLOWUPID>
<CREATEDBY>jsmith</CREATEDBY>
<status>Open</status>
<priority>Medium</priority>
<severity>Information</severity>
<category>Default</category>
project>Default
<mediatype>Voice</mediatype>
<visibility>Internal</visibility>
<DESCRIPTION>This customer downloaded the new driver XXXX v5.4 of the driver,
installed it and since then the printer doesn't work.</DESCRIPTION>
</ITEM>
<ITEM>
</ITEM>
</REPLY>
</WAPI>
```

See *Error Cases*, on page 92 for a list of errors to a unsuccessful reply.

Manipulating the Task Object

You can use the CRM API to add, modify, and get task data. This section outlines all the possible actions that can be performed on a task object and all the parameters that are used in both the request and the reply.

You can perform the following actions on the task object:

- Add
- Modify
- Get

Manipulating the Task Object

You can use the CRM API to add, modify, and get task data. This section outlines all the possible actions that can be performed on a task object and all the parameters that are used in both the request and the reply.

You can perform the following actions on the task object:

- Add
- Modify
- Get

Add Task

Add Task enables a tenant to add a new task to the CRM. This action corresponds to the "create" task action in the Agent Console Interface.

- "Task" keyword is not case sensitive.
- "Add" keyword is not case sensitive.
- Account Number is the mandatory parameter to pass.

Example 58 illustrates an add task operation.

Note: In Example 58, CF01 PICKLIST NAME and CF02.NAME are custom fields.

Example 58: Add Task operation

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</username>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Task" ACTION="Add">
<accountnum>10000019</accountnum>
<CREATEDBY>jsmith</CREATEDBY>
<SUBJECT>Phone Appointment</SUBJECT>
<DESCRIPTION>Customer has asked to discuss his questions about the
Promotional campaign.</DESCRIPTION>
<TASK STATUS>Pending</TASK STATUS>
<ASSIGNEDTO>jsmith</ASSIGNEDTO>
<CALL TYPE>DAA</CALL TYPE>
<MEDIA TYPE>Phone
<CAMPAIGNNAME>Promotional Campaign</CAMPAIGNNAME>
<CALLDURATION>360</CALLDURATION>
<TRANSID>1111111111
<EMAIL>jsmith@AcmeJets.com</EMAIL>
<MEDIATYPE>Phone</MEDIATYPE>
<CF01 PICKLIST.NAME>value1</CF01 PICKLIST.NAME>
<CF02.NAME>value2</CF02.NAME>
</COMMAND>
</WAPI>'
```

List of Mandatory and Optional Parameters

Table 11 summarizes the add case mandatory and optional parameters.

- M Mandatory
- O Optional

Table 11: Summary of Add Task parameters

Parameter	М	0	Format
ACCOUNTNUM	•		Account number of the customer who is related to the case. This account number is returned after a "GET" action on a "Customer" object.
SUBJECT		•	Any character
			Max - 255
			Invalid – whitespace only
DESCRIPTION		•	Any character
			Max - 2mb
			Invalid – whitespace only.
ASSIGNEDTO		•	The "assignedto" must be equal to a valid agents or groups user id. If "assignedto" is used, the "status" parameter must also be defined. If the value is not passed, the value of USERNAME is used.
			To search on tasks assigned to deleted agents, deleted groups, or unassigned agents the values are:
			"Deleted Agents", "Deleted Groups", or "None".
TASK_STATUS		•	The status must be one valid status (not deactivated) as defined in the Configuration Manager. If no status is provided, the default value is "New".
MEDIATYPE		•	The mediatype must be one valid media type (not deactivated) as defined in the Configuration Manager.
			Possible values are: "Phone", "Chat", "Email", "Support Center" or "None".
CALL_TYPE		•	Call Type should be valid.
PHONE		•	Max: 128 (digits and formatting characters used)
EMAIL		•	Valid email address, 255 characters long max.
DUEDATE			Due Date is the date and time by which the task should be completed. It is is represented as MMDDYYYYHH24MI (12 characters). Ex: 062120121315 for June 21st 2012 1:15PM
REMINDER		•	'Y' or 'N'. REMINDER needs to be used with DUEDATE
CAMPAIGNNAME		•	Any character; Max - 255; Invalid - Whitespace only

Parameter	М	0	Format
CALLDURATION		•	Call duration in seconds. Digits only; Max - 13
TRANSID		•	Transaction ID. Digits only; Max - 10
CF01_PICKLIST.NAME		•	Custom fields and their parameter names are added by the Administrator of the tenant. The values must be one valid type (not deactivated) as defined in the Configuration Manager. The list of possible values is defined by the Administrator of the tenant.
CF02.NAME		•	Custom fields and their parameter names are added by the Administrator of the tenant. The values must be one valid type (not deactivated) as defined in the Configuration Manager. The values passed should be in a text or number format.

Reply from CRM to Add a Task

Example 59 illustrates a reply to add a new task.

Example 59: Response to an Add Task operation.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<WAPI>
<REPLY STATUS="0" ERROR _STR="">
<TASKNUM>100</TASKNUM>
</REPLY>
</WAPI>
```

See *Error Cases*, on page 92 for a list of errors to a unsuccessful reply.

Modify Task

The Modify Task operation modifies an existing Internal CRM task record. This corresponds to the "edit" case action in the Agent Console Interface.

- "Modify" keyword is not case sensitive.
- The user may only send mandatory and modified parameters.
- Any parameter not supplied in a "Modify" request will retain the previous value.
- If no value is modified, a reply status of 0 will be returned but the task will be left unchanged
- Any change in the task updates its last modification date attribute and also causes the creation of a task activity that will report on the updated fields.

Example 60 illustrates a modify task request.

In the example, CF01_PICKLIST . NAME and CF02 . NAME identify previously identified custom Internal CRM fields.

Example 60: A ModifyTask operation that includes custom fields

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="Modify">
<TASKNUM>100</TASKNUM>
<SUBJECT>Phone call completed</SUBJECT>
<DESCRIPTION>Walk through on the promotional campaign options completed with the customer.</DESCRIPTION>
<TASK_STATUS>Closed</TASK_STATUS>
<ASSIGNEDTO>jsmith</ASSIGNEDTO>
<CF01_PICKLIST.NAME>value3</CF01_PICKLIST.NAME>
<CF02.NAME>value4</CF02.NAME>
</COMMAND>
<WAPI>'
```

List of Mandatory and Optional Parameters

summarizes the mandatory and optional modify task parameters

- M Mandatory
- O Optional

See Error Cases, on page 92 for a list of errors to a unsuccessful reply.

Table 12: Summary of Modify Task parameters

Parameter	М	0	Format
TASKNUM	•		Task number that needs to be modified. This task number is returned after a "GET" action on a "Task" object.
SUBJECT		•	Any character
			Max - 255
			Invalid – whitespace only
DESCRIPTION		•	Any character
			■ Max - 2mb
			■ Invalid – whitespace only.
ASSIGNEDTO		•	The "assignedto" must be equal to a valid agents or groups user id. If "assignedto" is used, the "status" parameter must also be defined. If the value is not passed, the value of USERNAME is used.
			To search on tasks assigned to deleted agents, deleted groups, or unassigned agents the values are:
			"Deleted Agents", "Deleted Groups", or "None".
TASK_STATUS		•	The status must be one valid status (not deactivated) as defined in the Configuration Manager.
MEDIATYPE		•	The mediatype must be one valid media type (not deactivated) as defined in the Configuration Manager.
			Possible values are: "Phone", "Chat", "Email", "Support Center" or "None".
CALL_TYPE		•	Call Type should be valid.
PHONE		•	Max: 128 (digits and formatting characters used)
EMAIL		•	Valid email address, 255 characters long max.
DUEDATE			Due Date is the date and the time by which the task should be completed. It is represented as MMDDYYYYHH24MI (12 characters). Ex: 062120121315 for June 21st 2012 1:15PM
REMINDER		•	'Y' or 'N'. REMINDER needs to be used with DUEDATE
CF01_PICKLIST.NAME		•	Custom fields and their parameter names are added by the Administrator of the tenant. The values must be one valid type (not deactivated) as defined in the Configuration Manager. The

Parameter	М	0	Format
			list of possible values is defined by the Administrator of the tenant.
CF02.NAME		•	Custom fields and their parameter names are added by the Administrator of the tenant. The values must be one valid type (not deactivated) as defined in the Configuration Manager. The values passed should be in a text or number format.

Reply from CRM to Modify a Task

Example 61 illustrates a reply to add a new case.

Example 61: Response to a Modify Task operation.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<WAPI>
<REPLY STATUS="0" ERROR _STR="">
<TASKNUM>100</TASKNUM>
</REPLY>
</WAPI>
```

See *Error Cases*, on page 92 for a list of errors to a unsuccessful reply.

Get Task

Get Task action enables a tenant to get a list of existing tasks from the CRM that matches the supplied parameter values. This action corresponds to the "Search", "View" and "List" actions in the Agent Console.

- "Get" keyword is not case sensitive.
- Command object only needs to be passed once, even when searching on multiple parameters.
- Parameter values being searched are case sensitive.
- The request returns entire descriptions for each task that matches the supplied parameter values, including the customer account number.

The query parameters on a "GET" request can vary in the following ways:

- 1. Query on a substring
- 2. Pass multiple parameters to generate an "OR" query
- 3. Pass multiple parameters to generate a "range" query

Tags that accept substring values

- Subject
- Phone
- Email
- CampaignName
- TransId

Note: Parameters values being searched as a substring require the use of "%". For example, to search on the word "call", the value should be set as: "%call%".

Tags that accept pre-defined values

- CreatedBy
- AssignedTo
- Task Status
- Call_Type
- Media_Type

Tags that can be paired for "range" queries

- FromCreateDate / ToCreateDate
- FromDueDate / ToDueDate
- FromLastModDate / ToLastModDate

NOTE: To retrieve tasks created on a particular date, "FromCreateDate" should correspond to the first desired date and the "ToCreateDate" should correspond to the day after. For e.g. a "FromCreateDate_" of '02292012' to a "ToCreateDate?" of '03012012' retrieves tasks created on the February 29th, 2012.

Examples of Get Queries

Example 62 illustrates a get case search given a case number.

Example 62: Get Task search by Task Number

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET"> <TASKNUM>100</TASKNUM>
</COMMAND>
</WAPI>'
```

The following example illustrates a get query to retrieve tasks whose status is closed. The task status must match the values defined in the Configuration Manager. The "GET" action on a Custom Field object can be used to determine the list of possible statuses.

Example 63: Get Task search by Task Status

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET">
<TASK_STATUS>Closed</TASK_STATUS>
</COMMAND>
</WAPI>'
```

The following example illustrates a search for retrieving tasks with a specified Due Date range.

Only one occurrence of the FROMDUEDATE and TODUEDATE parameters can be used in one GET request.

To retrieve tasks due on a particular day, "FromDueDate" should correspond to the first date desired and the "ToDueDate" should correspond to the day after. For example, a "FromDueDate" of '12072001' to a "ToDueDate" of '12082001' retrieves cases created on December 7th, 2001.

Example 64: Get Task search by Due Date

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET">
<FROMDUEDATE>060120120000</FROMDUEDATE>
<TODUEDATE>061520120000</TODUEDATE>
</COMMAND>
</WAPI>'
```

The following example illustrates a search for retrieving tasks assigned to a specific agent. You can pass multiple parameters to look for tasks assigned to multiple agents.

Example 65: Get Tasks assigned to an agent

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET">
<ASSIGNEDTO>jdoe</ASSIGNEDTO>
</COMMAND>
</WAPI>'
```

The following example illustrates a search for retrieving tasks created by a specific agent. You can pass multiple parameters to look for tasks created by multiple agents.

Example 66: Get Tasks created by an agent

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET">
<CREATEDBY>jdoe</CREATEDBY>
</COMMAND>
</WAPI>'
```

The following example illustrates a search for retrieving tasks with a specific keyword in the task subject.

Example 67: Get Tasks with a specific keyword in the subject

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET">
<SUBJECT>%promotional%</SUBJECT>
</COMMAND>
</WAPI>'
```

Note: Values being searched as a substring require the use of "%". For instance, to search for all promotional campaigns, the value should be set as: "%promotional%". If you don't use the '%' wildcard characters, the CRM API will perform an exact match instead as shown in the following example.

Example 68: Get Tasks with a specific keyword in the subject

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET">
<SUBJECT>promotional campaign 10</SUBJECT>
</COMMAND>
</WAPI>'
```

The following example illustrates a search for retrieving tasks last modified on a specific day.

Example 69: Get Tasks last modified on a specific date

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET">
<FROMLASTMODDATE>06202012</FROMLASTMODDATE>
<TOLASTMODDATE>06252012</TOLASTMODDATE>
</COMMAND>
</WAPI>'
```

Note: FROMLASTMODDATE and TOLASTMODDATE don't have to be used together necessarily. If only FROMLASTMODDATE is used, the CRM API engine will look for tasks modified after the passed date. Conversely, if only TOLASTMODATE is used, the CRM API engine will look for tasks modified before the passed date. The same concept applies for FROMCREATEDATE/TOCREATEDATE and for FROMDUEDATE/TODUEDATE.

The following example illustrates a search for retrieving tasks related to a campaign.

Example 70: Get Tasks related to a campaign

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET">
<CAMPAIGNNAME>%Promotional campaign for Holidays%</CAMPAIGNNAME>
</COMMAND>
</WAPI>'
```

The following example illustrates a search for retrieving tasks related to a specific phone number.

Example 71: Get Tasks targeted to a specific phone

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET">
<PHONE>1-555-292-8608</PHONE>
</COMMAND>
</WAPI>'
```

Example 72: Get Tasks targeted to a specific Email

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET">
<EMAIL>sirben.kingsley@stars.com</EMAIL>
</COMMAND>
</WAPI>'
```

Example 73: Get Tasks search on Transaction ID

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET">
<TRANSID>12345</TRANSID>
</COMMAND>
</COMMAND>
</WAPI>'
```

Example 74: Get Tasks search on Call Type

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET">
<CALL_TYPE>Outbound</CALL_TYPE>
</COMMAND>
</WAPI>'
```

Example 75: Get Tasks search on Media Type

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET">
<MEDIA_TYPE>Outbound</MEDIA_TYPE>
</COMMAND>
</WAPI>'
```

The next example illustrates a Get Task query without passing any parameters. The CRM API returns:

- FIRSTID: ID of first task record available
- LASTID: ID of last task record available
- Total: total number of task records

Example 76: Get Tasks search without passing parameters

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET">
</COMMAND>
</WAPI>'
```

Reply received from CRM:

Example 77: Reply for a Get Task search without passing parameters

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<WAPI>
<REPLY STATUS="0" ERROR_STR="0">
<ITEM> <FIRSTID>170</FIRSTID>
<LASTID>263</LASTID>
<TOTAL>79</TOTAL>
</ITEM>
```

The following example retrieves tasks matching a specific value of a custom picklist field.

Example 78: Get Tasks search on a custom task picklist field

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET">
<TASK_PICKLIST>two</TASK_PICKLIST>
```

```
</COMMAND>
</WAPI>'
```

Note: "TASK_PICKLIST" is the "Field Name" attribute for the picklist custom field defined in the CRM. You have to pass a valid value.

The following example gets tasks with a specific value in a custom task text field.

Example 79: Get Tasks search on a text custom field value

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET">
<TASK_TEXT>random text</TASK_TEXT>
</COMMAND>
</WAPI>'
```

Note: "TASK_TEXT" is the "Field Name" attribute for the text custom field defined in the CRM. You have to pass a valid value.

The following example retrieves tasks matching the date specified in the Task_Date custom field.

Example 80: Get Tasks search on a date custom field value

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET">
<TASK_DATE>06222012</TASK_DATE>
</COMMAND>
</WAPI>'
```

Note: "TASK_DATE" is the "Field Name" attribute for the date custom field defined in the CRM.

The following example gets tasks created by and assigned to deleted agents

Example 81: Get Tasks search on a date custom field value

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Case" ACTION="GET">
<CREATEDBY>Deleted Agents</CREATEDBY>
<ASSIGNEDTO>Deleted Agents</ASSIGNEDTO>
</COMMAND>
</WAPI>'
```

Note: Deleted Agents" is the value the CRM API user has to specify to fetch tasks created by deleted agents. This value is not case sensitive. Only one space character should separate "Deleted" and "Agents".

List of Parameters

summarizes the get task parameters.

At least one parameter must be used in a GET request.

Table 13: Summary of Get Task parameters

Parameter	Format
TASKNUM	Task number that needs to be modified. This task number is returned after a "GET" action on a "Task" object.
SUBJECT	Any character
	Max - 255
	Invalid – whitespace only
DESCRIPTION	Any character
	■ Max - 2mb
	■ Invalid – whitespace only.
ASSIGNEDTO	The "assignedto" must be equal to a valid agents or groups user id. If "assignedto" is used, the "status" parameter must also be defined. If the value is not passed, the value of USERNAME is used.
	To search on tasks assigned to deleted agents, deleted groups, or unassigned agents the values are:
	"Deleted Agents", "Deleted Groups", or "None".
CREATEDBY	Should be a valid agent, or "Deleted Agents" to search on tasks created by deleted agents.
TASK_STATUS	The status must be one valid status (not deactivated) as defined in the Configuration Manager.
FROMDUEDATE	The format for this attribute is: MMDDYYYY where MM is the month, DD the day and YYYY the year.
TODUEDATE	The format for this attribute is: MMDDYYYY where MM is the month, DD the day and YYYY the year.
FROMLASTMODDATE	The format for this attribute is: MMDDYYYY where MM is the month, DD the day and YYYY the year.
TOLASTMODDATE	The format for this attribute is: MMDDYYYY where MM is the month, DD the day and YYYY the year.
MEDIATYPE	The mediatype must be one valid media type (not deactivated) as defined in the Configuration Manager.
	Possible values are: "Phone", "Chat", "Email", "Support Center" or "None".
CALL_TYPE	Call Type should be valid.

Parameter	Format				
PHONE	Max: 128 (digits and formatting characters used)				
EMAIL	Valid email address, 255 characters long max.				
CAMPAGINNAME	Any character; Max - 255; Invalid - Whitespace only				
TRANSID	Transaction ID. Digits only; Max - 10.				
REMINDER	'Y' or 'N'. REMINDER needs to be used with DUEDATE				
CF01_PICKLIST.NAME	Custom fields and their parameter names are added by the Administrator of the tenant. The values must be one valid type (not deactivated) as defined in the Configuration Manager. The list of possible values is defined by the Administrator of the tenant.				
CF02.NAME	Custom fields and their parameter names are added by the Administrator of the tenant. The values must be one valid type (not deactivated) as defined in the Configuration Manager. The values passed should be in a text or number format.				

Reply from CRM to Get Task

Example 82 illustrates a to a request for an existing case.

In the preceding example, task_text, task_text_unique, task_url, task_number, task_number_unique, task_date and task_picklist are custom fields with the obvious type.

Example 82: Reply to successful Get Task Operation

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<WAPI>
<REPLY ERROR CODE="0" ERROR STR="" STATUS="0">
<ITEM>
<TASKNUM>183</TASKNUM>
<SUBJECT>Follow-up with customer on pricing</SUBJECT>
<DESCRIPTION>Description for task/DESCRIPTION>
<CREATEDDATE>04062012 18:33:49</CREATEDDATE>
<DUEDATE>06092012 03:00:00
<REMINDER>N</REMINDER>
<PHONE/>
<LASTMODDATE>04092012 18:04:02
<EMAIL>angel@8x8.com</EMAIL>
<TRANSID/>
<CALLDURATION>45</CALLDURATION>
<CAMPAIGNNAME/>
<task text>some text</task text>
<task text unique>some unique string 4724732827489324893274</task text unique>
<task url>http://www.8x8.com</task url>
<task number>111444</task number>
<task number unique>613293374</task number unique>
<task date>04062012</task date>
<task picklist>three</task picklist>
<task_status>New</task_status>
<media_type>Email</media_type>
<call_type/>
<accountnum>10000001</accountnum>
<ASSIGNEDTO>pma</ASSIGNEDTO>
<CREATEDBY>pma</CREATEDBY>
</ITEM>
<ITEM>
</ITEM>
</REPLY>
</WAPI>
```

See Error Cases, on page 92 for a list of errors to a unsuccessful reply.

Note

- When ASSIGNEDTO is not passed in a GET TASK query, only tasks assigned to valid (not deleted) agents will be returned.
- When CREATEDBY is not passed in a GET TASK query, only tasks created by valid agents (not deleted agents) will be returned.
- So, if you want to retrieve a task created AND assigned to deleted agents (not necessarily the same as long as they are both deleted), you will have to use both tags in the XML query with the same "Deleted Agents" value.

Manipulating the FAQCATEGORY Object

The Contactual CRM allows tenant administrator or Supervisor to create FAQ library for its agents to quickly find answers to frequently asked questions. The FAQ library contains FAQ categories and organizes FAQs and answers under each category.

For example, AcmeJets defines FAQ categories for Support, Sales, and Safety. Questions and answers are grouped under the related category.

This section outlines the actions that can be performed on an FAQ category object and the parameters that are used in both the request and the reply.

For the FAQ category object, the following actions can be performed:

- Add
- Modify
- Get
- Delete
- List

Add FAQCATEGORY

This action enables a tenant to add a new FAQ category to the CRM. This action corresponds to the "Create" action in the Agent Supervisor Interface.

- FAQcategory keyword is not case sensitive.
- "Add" keyword is not case sensitive.

Example 83 illustrates an add FAQCATEGORY operation.

Example 83: Example of an Add FAQCATEGORY operation

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="FAQCategory" ACTION="Add">
<FAQCATEGORY>Safety</FAQCATEGORY>
<PRIVATE>Y</PRIVATE>
</COMMAND>
</WAPI>'
```

Mandatory and Optional Parameters

Table 14 list the mandatory and optional parameters.

- M Mandatory
- O Optional

Table 14: Summary of Add FAQCATEGORY parameters

Parameter	М	0	Format
FAQCATEGORY	•		Any character not defined invalid Max - 50 Invalid – White space only
PRIVATE		•	Possible values: Y Or N. Values are case sensitive. Selecting Y declares the FAQ category private and vice versa. The default value is N (no).

Reply from CRM to Add FAQCATEGORY

Example 84 illustrates the respone to an Add FAQCATEGORY request.

Example 84: Successful response to Add FAQCATEGORY

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<WAPI>
<REPLY ERROR_CODE="0" ERROR_STR="" STATUS="0">
<FAQCATID>1</FAQCATID>
</REPLY>
</WAPI>
```

See *Error Cases*, on page 92 for a list of errors to a unsuccessful reply.

Modify FAQCATEGORY

This action enables a tenant to modify an existing FAQcategory in the CRM. This action corresponds to the "edit FAQ category" action in the Agent Supervisor Interface.

Example 85 illustrates a Modify FAQCATEGORY operation.

- Modify keyword is not case sensitive.
- Any parameter not supplied in a "Modify" request retains its previous value.
- The parameter value passed to declare a category private or public is case sensitive.

The example below illustrates a request to modify an existing FAQ category. The category is referenced by its category ID, which can be accessed by:

- passing a request to get FAQ categories OR
- passing a request to list all FAQ categories defined for the tenant

Example 85: x

```
<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="FAQCategory" ACTION="Modify"> <FAQCATID>1</FAQCATID>
<FAQCATEGORY>Legalities</FAQCATEGORY>
<PRIVATE>Y</PRIVATE>
</COMMAND>
</WAPI>
```

Summary of Mandatory and Optional Parameters

Table 15 lists the mandatory and optional Modify FAQCATEGORY parameters.

- M Mandatory
- O Optional

Table 15: Summary of Modify FAQCATEGORY parameters

Parameter	М	0	Format
Parameter	М	0	Format
FAQCATID	•		Internal ID of the Category. This ID is returned after an "Add" action or a "Get" action on a FAQCATEGORY Object.
FAQCATEGORY	•		Any character
			Max - 50
			Invalid - Whitespace only
PRIVATE		•	Possible values: Y or N. Values are case sensitive.
			The default value is N.

See Error Cases, on page 92 for a list of errors to a unsuccessful reply.

Reply from CRM to Modify FAQCATEGORY

Example 86 illustrates a reply to a request for modification of a FAQCATEGORY.

Example 86: Reply to Modify FAQCATEGORY

```
<?xml version="1. 0" encoding="ISO-8859-1"?>
<WAPI>
<REPLY ERROR_CODE="0" ERROR_STR="" STATUS="0">
<FAQCATID>1</FAQCATID>
</REPLY>
</WAPI>
```

See *Error Cases*, on page 92 for a list of errors to a unsuccessful reply.

Get FAQCATEGORY

Get FAQCATEGORY retrieves all FAQ Categories from the Internal CRM.

Example 87 illustrates a request for all FAQ Categories from the Internal CRM.

- Get keyword is not case sensitive.
- Command object only needs to be passed once, even when searching on multiple parameters.
- Parameter values being searched are case sensitive.

The following example fetches all FAQ categories defined in the tenant.

Example 87: Example of a Get FAQCATEGORY operation

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="FAQCategory" ACTION="Get">
</COMMAND>
</WAPI>'
```

Reply from CRM to Get FAQCATEGORY

Example 88 illustrates a reply request for a an existing FAQCATEGORY.

Example 88: Reply to Get FAQCATEGORY

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<WAPI>-<REPLY ERROR_CODE="0" ERROR_STR="" STATUS="0">-
<ITEM><FAQCATID>2</FAQCATID><FAQCATEGORY>Safety</FAQCATEGORY></ITEM>
<ITEM><FAQCATID>3</FAQCATID><FAQCATEGORY>legalities</FAQCATEGORY></ITEM>
<ITEM><FAQCATID>1</FAQCATID><FAQCATEGORY>Services</FAQCATEGORY></ITEM>
</REPLY>
</WAPI>
```

See *Error Cases*, on page 92 for a list of errors to a unsuccessful reply.

Delete FAQCATEGORY

Example 89 illustrates a request to delete an existing Internal CRM.

The "Delete" keyword is not case sensitive.

Example 89: Delete FAQCATEGORY request

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="FAQCategory" ACTION="Delete"> <FAQCATID>4</FAQCATID>
</COMMAND>
</WAPI>'
```

Mandatory and Optional Parameters

Table 16 summarizes the mandatory and optional Delete FAQCATEGORY parameters.

- M Mandatory
- O Optional

Table 16: Summary of Delete FAQCATEGORY parameters

Parameter	М	0	Format
FAQCATID	•		Internal ID of the Category. This ID is returned after an "Add" action or a "Get" action on a FAQCATEGORY Object.

Reply from CRM to Delete FAQCATEGORY

Example 90 illustrates a request to delete an existing FAQCATEGORY.

- You cannot delete a category that contains FAQ items.
- You must delete all FAQ items in a category before attempting to delete a FAQCATEGORY.

Example 90: A Delete FAQCATEGORY operation

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<WAPI>
<REPLY ERROR_CODE="0" STATUS="0" ERROR_STR="">
</REPLY>
</WAPI>
```

See Error Cases, on page 92 for a list of errors to a unsuccessful reply.

Manipulating the FAQ Object

This section outlines the possible actions that can be performed on an FAQ object and the parameters that are used in both the request and the reply.

You can perform the following actions on the FAQ object:

- Add
- Modify
- Get
- Delete

Add FAQ

This action enables a tenant to add an FAQ to a FAQCATEGORY in the Internal CRM. This corresponds to the "add" FAQ action in the Agent Console Interface.

Example 91 illustrates an Add FAQ request.

- FAQ keyword is not case sensitive.
- FAQs are limited to non-HTML format only.
- FAQCATID identifies the FAQ category to which the FAQ question and answer are added.
- FAQCATID, Question and Answer are mandatory data to add a fag item.
- SHORTCUT uniquely identifies a FAQ item and allows agents to reference and retrieve the item quickly without having to search the FAQ library.
- PRIVATE declares a faq item private or public. The possible values or Y or N. The values are case sensitive.

Example 91: An Add FAQ operation

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>wapi</USERNAME>
<PASSWORD>wapi1</PASSWORD>
<COMMAND OBJECT="FAQ" ACTION="Add">
<FAQCATID>1</FAQCATID>
<QUESTION>What types of private jets are available for leasing?</QUESTION>
<ANSWER> Private jets are typically categorized into three different size classes ¬ light jets, mid-sized jets and heavy jets. Light Jets can seat up to 4 passengers, mid-sized jets seat up to 8, while the heavy jets seat up to 12
passengers.</ANSWER>
<SHORTCUT>Jettype1</SHORTCUT>
<PRIVATE>Y</PRIVATE>
</COMMAND>
</WAPI>'
```

Summary of Mandatory and Optional Parameters

Table 17 summarizes the mandatory and optional parameters for the Add FAQ operation.

- M Mandatory
- O Optional

Table 17: Summary of Add FAQ parameters

Parameter	М	o	Format
FAQCATID	•		Internal ID of the Category. This ID is returned after an "Add" action or a "Get" action on a FAQCATEGORY Object.
QUESTION	•		Any character Max - 250 Invalid – White space only
ANSWER	•		Any character Max - 4000 Invalid – White space only
SHORTCUT		•	Any character Max - 20 Invalid – White space only
PRIVATE		•	Possible values: Y or N. Values are case sensitive. The default value is N.

Reply from CRM to Add FAQ

Example 92 illustrates a reply to an add FAQ request.

Example 92: An Add FAQ operation

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<WAPI>
<REPLY STATUS_CODE="0" ERROR STR="" STATUS="0">
<FAQID>1</FAQID>
</REPLY>
</WAPI>
```

See Error Cases, on page 92 for a list of errors to a unsuccessful reply.

Modify FAQ

The Modify FAQ action allows you to update an existing FAQ item in a category.

- Modify keyword is not case sensitive.
- Any parameter not supplied in a "Modify" request retains the previous value.
- FAQID uniquely identifies the FAQ item to be updated and hence is mandatory.

Example 93 illustrates a Modify FAQ request.

Example 93: Modfiy FAQ request

```
'<WAPI>
<TENANT>AcmeJets</fr>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="FAQ" ACTION="Modify">
<FAQID>1</FAQID>
<QUESTION>What types of private jets are available for leasing?</QUESTION>
<ANSWER>Private jets are typically categorized into three different size classes light jets, mid-sized jets and heavy jets. Light Jets can seat up to 4 passengers, mid-sized jets seat up to 8, while the heavy jets seat up to 12
passengers.</ANSWER>
</COMMAND>
</WAPI>'
```

Summary of Mandatory and Optional Parameter

Table 18 summarizes the mandatory and optional Modify FAQ parameters

- M Mandatory
- O Optional

Table 18: Summary of Modify FAQ parameterrs

Parameter	М	o	Format
FAQID	•		Internal ID of the FAQ. This ID is returned after an "Add" action or a "Get" action on a FAQ Object.
QUESTION		•	Any character
			Max - 250
			Invalid – White space only
			Note: If the field is not specified, the database field will be created empty.
ANSWER		•	Any character
			Max - 4000
			Invalid – White space only
			If the field is not specified, the database field will be created empty.
SHORTCUT		•	Any character
			Max - 20
			Invalid – White space only
PRIVATE		•	Possible values: Y or N. Values are not case sensitive.
			The default value is N.

Reply from CRM to Modify FAQ

Example 94 illustrates a reply to a modify FAQ request.

Example 94: Response to a Modify FAQ request

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<WAPI>
<REPLY ERROR_CODE="0" ERROR_STR="" STATUS="0">
<FAQID>1</FAQID>
</REPLY>
</WAPI>
```

See Error Cases, on page 92 for a list of errors to a unsuccessful reply.

Get FAQ

Get FAQ lists all FAQs in a specified FAQ category. It also enables the tenant to search for all FAQs containing a specified sub-string within the QUESTION or ANSWER parameters.

- Get keyword is not case sensitive.
- Command object only needs to be passed once, even when searching on multiple parameters.
- Parameter values being searched are case sensitive.

GET Options

- List all FAQs in an FAQ Category
- List all FAQs containing a specified sub-string

Example 95 lists all FAQs in an FAQ Category 100.

Example 95: List all FAQs from a FAQ category

```
'<WAPI>
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="FAQ" ACTION="Get">
<FAQCATID>1</FAQCATID>
</COMMAND>
</WAPI>'
```

Example 96 lists all FAQs that use wild card characters to return all FAQS that contain the substring light jets.

Example 96: List all FAQS that contain the substring DMV Office

```
'<WAPI>
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="FAQ" ACTION="get">
<STRING>%light jets%</STRING>
</COMMAND>
</WAPI>'
```

Note: Use <STRING>%<STRING> to fetch all FAQs in the FAQ library.

List of Parameters

Table 19 summarizes the Get FAQ parameters.

You can search for FAQ based on FAQCATID and STRING parameters only. The STRING parameter looks for a specified string in QUESTION or ANSWER fields.

Table 19: Summary of Get FAQ parameters

Parameter	Format				
FAQCATID	Internal ID of the FAQ. This ID is returned after an "Add" action or a "Get" action on a FAQ Object.				
STRING	Any character				

Reply from CRM to Get FAQ

Example 97 illustrates a reply to a Get FAQ request.

Example 97: Response to Get FAQ

```
<?xml version="1.0" encoding="ISO-8859-1"?>
-<WAPI>
-<REPLY ERROR_CODE="0" ERROR_STR="" STATUS="0">
-<ITEM>
<FAQCATID>1</FAQCATID>
<FAQID>3</FAQID>
<QUESTION>What types of private jets are available for leasing?</QUESTION>
<ANSWER>Private jets are typically categorized into three different size classes
iż½ light jets, mid-sized jets and heavy jets. Light Jets can seat up to 4
passengers, mid-sized jets seat up to 8, while the heavy jets seat up to 12
passengers.</ANSWER>
<SHORTCUT>Jettype1</shortcut>
<PRIVATE>Y</PRIVATE>
</ITEM>
<ITEM>
</ITEM></REPLY></WAPI>
```

See *Error Cases*, on page 92 for a list of errors to a unsuccessful reply.

Delete FAQ

Delete FAQ deletes a FAQ from a FAQCategory.

- Delete keyword is not case sensitive.
- FAQID is mandatory to refer to a FAQ item to be deleted.

Example 98 illustrates a Delete FAQ request.

Example 98: A Delete FAQ operation

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="FAQ" ACTION="Delete">
<FAQID>45</FAQID>
</COMMAND>
</WAPI>'
```

Summary of Parameters

Table 20 summarizes the Delete FAQ parameter.

Table 20: Summary of Delete FAQ parameter

Parameter	М	0	Format
FAQID	•		Internal ID of the FAQ. This ID is returned after an "Add" action or a "Get" action on a FAQ Object.

Reply from CRM to Delete FAQ

Example 99 illustrates a reply to request to delete a FAQ.

Example 99: Resopnse to Delete FAQ request

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<WAPI>
<REPLY ERROR_CODE="0" ERROR_STR="" STATUS="0"/>
</WAPI>
```

See Error Cases, on page 92 for a list of errors to a unsuccessful reply.

Manipulating the Custom Field Objects and Values

This section outlines the actions that can be performed on a Custom Field object and Custom Field values and all the parameters that are used in both the request and the reply.

- For Custom Field objects, the CRM API supports the **GET** operation.
- For Custom Field values, the CRM API supports GET and MODIFY

Get Custom Fields

This action allows a tenant to get the different values of an existing "pick list" custom field in the CRM database.

- Get keyword is not case sensitive.
- Command object only needs to be passed once, even when searching on multiple parameters.
- Parameter values being searched are case sensitive.
- The contact center administrator uses the Configuration Manager to define custom fields and values.

Example 100 illustrates a Get Custom Field request. In this example, the pick list values of the custom field by name *PrivacyLevel* are returned.

Example 100: Get Custom Field request

```
'<WAPI>
'<WAPI>
<TENANT>AcmeJets</free/
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="Customfield" ACTION="GET"> <NAME>PrivacyLevel</NAME>
</COMMAND>
</WAPI>'
```

List of Parameters

Table 21 summarizes the Get Custom Field parameters.

At least one CUSTOM FIELD parameter must be used in a "GET" request.

Table 21: Summary of Get Custom Field parameter

Parameter	Format
Custom Field Name	Name of the pick list custom field.
	The contact center administrator uses the Configuration Manager to define custom fields and values.

Reply from CRM to Get Custom Fields

Example 101 illustrates a reply to a Get Custom Field request.

If the parameter given in the "GET" request doesn't correspond to a pick list, the Internal CRM returns an empty response.

Example 101: Illustration of response to Get Custom Fields operation

```
<?xml version="1.0" encoding="ISO-8859-1"?>
-<WAPI>-<REPLY ERROR_CODE="0" ERROR_STR="" STATUS="0">
-<ITEM><VALUE>NO Solicitation</VALUE></ITEM>
-<ITEM><VALUE>Unknown</VALUE></ITEM>
-<ITEM><VALUE>Neutral</VALUE></ITEM>
</REPLY></WAPI>
```

See *Error Cases*, on page 92 for a list of errors to a unsuccessful reply.

Modify Custom Field Values

This action allows a tenant to modify custom field values for custom fields that the tenant administrator has added in the Configuration Manager. Custom fields can be added to the Customer, Case, and Followup sections of the Agent Console Interface.

- Custom fields can be added to the Customer, Case, and Followup objects of the CRM.
- Only custom field values can be modified, not custom field objects.
- Modify keyword is not case sensitive.
- Any parameter not supplied in a "Modify" request will retain the previous value.
- Custom fields can vary for each tenant
- When submitting a "Modify" request for a Customer or Case object, you must ensure the mandatory parameters for this type of object are supplied.

A list of the mandatory and optional parameters for a Customer or Case object can be found in their respective chapters in this document.

Example 102 modifies the custom field values CF0 1 PICKLIST . NAME and CF02 . NAME.

Example 102: A modify custom field values request

```
'<wapracerous'
'<tenant>acmejets</fenant>
<username>johnd</username>
<username>johnd123</password>
command object="customer" action="modify">
<command object="customer" action="modify">
<accountnum>1000019</accountnum>
<firstname>bill</firstname>
<username>cuastname>gates</username>
<username>cuserous'
<username>cu
```

List of Mandatory and Optional Parameters

Table 22 summarizes the mandatory and optional Modify Custom Field parameters

- M Mandatory
- O Optional

Any parameter not supplied in a "Modify" retains its previous value.

Table 22: Summary of Modify Custom Field parameters

Parameter	М	0	Format
ACCOUNTNUM	•		Customer account number to be modified (ACCOUNTNUM as returned by GET request).
FIRSTNAME		•	Character Max - 30
LASTNAME		•	Character Max - 30
CF01_PICKLIST.NAME		•	Custom fields and their parameter names are added by the Administrator of the tenant. The values must be valid (not deleted) and match the values as defined in the Configuration Manager. The tenant administrator defines the custom field values.
CF02.NAME		•	Custom fields and their parameter names are added by the Administrator of the tenant. The values must be valid (not deleted) as defined in the Configuration Manager. The values passed should be in a text or number format.

Reply from the CRM to Modify Custom Field Values

Example 103 illustrates a reply to a request to modify custom field values.

Example 103: Response to Modify Custom Field request

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<WAPI>
<REPLY STATUS="0" ERROR_STR="">
<ITEM>
<ACCOUNTNUM>100019</ACCOUNTNUM>
</ITEM>
</REPLY>
</WAPI>
```

See Error Cases, on page 92 for a list of errors to a unsuccessful reply.

Listing the Attributes Values

This section provides examples on how to list attributes.

List COUNTRY Attributes

Example 104 illustrates a request to list the COUNTRY attribute.

Example 104: Example of a list COUNTRY operation

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="COUNTRY" ACTION="LIST">
</COMMAND>
</WAPI>'
```

Reply from CRM to List COUNTRY

Example 105 illustrates a reply to the request to list the COUNTRY attribute.

Example 105:

```
..
<?xml version="1.0" encoding="ISO-8859-1"?>
<WAPI>
<REPLY ERROR_CODE="0" ERROR_STR="" STATUS="0">
<ITEM>
<COUNTRY>Afghanistan</COUNTRY>
</ITEM> <ITEM>
<COUNTRY>Albania</COUNTRY>
</ITEM> <ITEM>
<COUNTRY>Albania</COUNTRY>
</ITEM> <ITEM>
</ITEM> <ITEM>
</ITEM> <ITEM>
<COUNTRY>Algeria</COUNTRY>
</ITEM> <ITEM>
</COUNTRY>Algeria</COUNTRY>
</ITEM> <ITEM> ...
</ITEM> </REPLY> </WAPI>
```

See *Error Cases*, on page 92 for a list of errors to a unsuccessful reply.

Listing Tenant Configuration Attributes

This section describes how to retrieve tenant configuration attributes, such as Groups of Agents and Agent Names.

List Agent GROUPS

Example 106 illustrates a request to list the current agent GROUPS of the tenant.

Example 106: A request to list agent GROUPS

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="GROUPS" ACTION="LIST">
</COMMAND>
</WAPI>'
```

Reply to List Agent GROUPS

Example 107 illustrates a request to list agent groups.

Example 107: A request to list agent GROUPS

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<WAPI>
<REPLY STATUS="0" ERROR_STR="">
<ITEM> <AGENTID>bobbp</AGENT ID>
<AGENT>bob power< /AGENT>
</ITEM>
<ITEM>
<ITEM>
<AGENTID>johnd</AGENTID>
<AGENTID>johnd doe</AGENT>
</ITEM>
<ITEM>
</ITEM>
```

See Error Cases, on page 92 for a list of errors to a unsuccessful reply.

List AGENTS

Example 108 illustrates a list AGENTS request.

Example 108: A list AGENTS request.

```
'<WAPI>
<TENANT>AcmeJets</TENANT>

<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="AGENTS" ACTION="LIST">
</COMMAND>
</WAPI>'
```

Reply to List AGENTS

Example 109 illustrates the reply to a list AGENTS request.

Example 109: Response to a list AGENTS request.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<WAPI>
<REPLY ERROR_CODE="0" ERROR_STR="" STATUS="0">
<ITEM>
<AGENTID>bobbp</AGENT ID>
<AGENT>bob power</AGENT>
</ITEM>
<ITEM>
<AGENTID>johnd</AGENTID>
<AGENTID>johnd</AGENTID>
<AGENT>john doe</AGENT>
</ITEM>
<ITEM>
<ITEM>
<ITEM>
<ITEM>
<ITEM>
<ITEM>
</ITEM>
</ITEM>
</ITEM>
</REPLY>
</WAPI>
```

See *Error Cases*, on page 92 for a list of errors to a unsuccessful reply.

List FAQ Categories

Example 110 illustrates a list FAQ categories defined in the tenant.

Example 110: A list FAQ Categories request.

```
'<WAPI>
<TENANT>AcmeJets</TENANT>
<USERNAME>johnd</USERNAME>
<PASSWORD>john123</PASSWORD>
<COMMAND OBJECT="FAQCATEGORY"" ACTION="LIST">
</COMMAND>
</WAPI>'
```

Reply to list FAQ Category

Example 111 illustrates the reply to a list FAQ categories.

Example 111: Response to a list AGENTS request.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
-<WAPI>-<REPLY ERROR_CODE="0" ERROR_STR="" STATUS="0">
-<ITEM><FAQCATID>2</FAQCATID>
<FAQCATEGORY>Safety</FAQCATEGORY>
</ITEM>
-<ITEM><FAQCATID>4</FAQCATID>
<FAQCATEGORY>legal</FAQCATEGORY></ITEM>
-<ITEM>
FAQCATID>1</FAQCATID>
<FAQCATID>1</FAQCATID>
<FAQCATID>1</FAQCATID>
<FAQCATID>1</FAQCATID>
<FAQCATID>1</FAQCATID>
<FAQCATEGORY>Services</FAQCATEGORY>
</ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></ITEM></IT
```

See Error Cases, on page 92 for a list of errors to a unsuccessful reply.

Historical Statistics

Beginning in release 3.0.0, the Internal CRM API no longer supports retrieval of Historical Statistics.

However, to export historical statistics to Microsoft Excel in the Agent Supervisor Console, in **Reporting**, **Historical Reports**.

Error Cases

Table 23lists the error messages returned in response to improperly formed queries to the Internal CRM.

Table 23: List of CRM API error messages

Error Message	Error Definition
Action 'Action Name' Not Valid for the Object 'Object Name'	The action value for the given object is not valid.
Action Value not passed	The action value was not passed in the XML string.
Action Value Not Valid	The value of the ACTION in the XML string is not valid
CASENUM 'case number' is an EMAIL Case. The Description of an Email Case cannot be modified	Case created via an email to tenant.Thedescription of email cases cannot be modified.
Duplicate Tag 'Alias Name'	Duplicate tags are passed for the ADD, MODIFY or DELETE actions.
Duplicate value 'Value passed' of 'Alias Name'	The value inserted should be unique and the value passed already exists in the database.
FAQCATID not passed	FAQ category is deleted and FAQCATID tag is not passed.
FAQ Category cannot be deleted when FAQ exists for the given category	FAQ category cannot be deleted when a FAQ exists in the given category.
Inconsistent tags. AUTOPASSWORD is TRUE and PASSWORD is also provided	AUTOPASSWORD TRUE means that password will be automatically generated. Therefore there is a conflict when the password is also provided.
Input Error: The tag 'tag name' not passed	An expected tag is not passed.
Internal Error	Internal error. Please contact customer support.
No tags are found	No tags passed in the ADD, MODIFY or DELETE action.
None of the possible mandatory tags or its values is found. Please check the documentation and pass at least one mandatory tag	Mandatory tag not passed.
Not a valid XML string	XML string not valid.
Not a valid XML string. Error after 'Tag Name'	XML string not valid; error after 'Tag Name'.

Error Message	Error Definition
OBJECT name 'Object Name' not valid	Object name not valid.
OBJECT 'Object Name' is not a valid object	The value of the Object in the XML string is not valid.
OBJECT value not passed	In the XML string passed there is no object value.
Regular expression not allowed for the alias 'Alias Name'	For GET Action, some tag values can be passed with% and for other tag values it cannot be passed.
TAGS Incompatible. ASSIGNEDTO passed but no STATUS is passed. By default STATUS value is label with USABILITY ='UNA'	Must pass the ASSIGNEDTO and STATUS parameters. By default, the status is set to unassigned.
TAGS Incompatible. ASSIGNEDTO passed but STATUS is UNASSIGNED	ASSIGNEDTO parameter passed but status is set to unassigned.
TAGS Incompatible. ASSIGNEDTO passed with STATUS = Unassigned	Case cannot be assigned to an agent when the STATUS of the case is unassigned.
TAGS Incompatible. EMPTY ASSIGNEDTO passed with STATUS = OPEN	Case must be assigned to an agent if the STATUS is open.
TAGS Incompatible. The usability of STATUS cannot be OPN, when the Case is not assigned to some agent	The status of the case cannot be open when the case is not assigned to an agent. Also, possible that the ASSIGNEDTO tag was not passed.
TAGS Incompatible. The usability of STATUS cannot be UNA, when the Case is assigned to some agent	The status of the case is changed to Unassigned without passing ASSIGNEDTO tag.
TAGS Incompatible. The usability of STATUS is OPN, but no	ASSIGNEDTO parameter needs to be passed in XML string.
ASSIGNEDTO passed TAGS Incompatible. The value of STATUS cannot be Unassigned, when the Case is assigned to some agent	The status cannot be unassigned when the case is assigned to an agent.
The ALIAS 'AUTOPASSWD' is not given	AUTOPASSWD tag is not passed in "add customer" request.
The ALIAS 'AUTOPASSWD' is FALSE, but ALIAS 'PASSWORD' is not provided	IfAUTOPASS WORDisfalse,thealias PASSWORD must be provided.
The length of the value for the alias 'Alias Name' doesn't confirm to the criterion. The length of the value passed is 'length of the value', min length allowed is 'min length allowed' and max length allowed is 'max length allowed'	Length mismatch. Consult documentation

Error Message	Error Definition
The length of the value of the tag 'Alias Name' is 'Passed Length' which is greater than the allowed value 'Allowed Max Value	Custom field length exceeds the maximum value.
The mandatory tag 'Alias Name' is not passed or the value for this tag is not passed	The case number is not valid.
The passed caseid 'Case Number' is not valid	The value passed for the given alias doesn't meet the specified criterion.
The passed value 'Alias Value' of the alias 'Alias Name' doesn't meet the specified criterion	The tag is not a valid action on the object type selected.
The tag 'Alias Name' is not a valid tag for the 'Action' on the 'object'	Custom field which status=deleted.
The STATUS of the tag 'Alias Name' is 'DEL'. No action allowed on this tag	A case is assigned to a deleted agent/group.
The STATUS of the value 'Value of the ASSIGNEDTO' of the ASSIGNEDTO tag is deleted	Value passed for ASSIGNEDTO is deleted.
The STATUS of the value 'Value Passed' of the ASSIGNEDTO tag is deleted	Custom field of date data type is not in the correct format.
The value of the passed alias 'Alias Name' is not in the correct format (mmddyyyy)	The case number is not valid.
The value 'Passed Value' of the alias 'Alias Name' is not valid	The 'Value' for the tag 'Tag Name' is not valid.
The value 'value' for the tag 'tag name' is not valid	The 'Value passed' for the tag 'Tag Name' is not valid.
The value = 'Value passed' of the tag 'Tag Name' is not valid	XML string not correct.
XML string not correct. Error at line 'line number': 'Error Message'	The 'Value' for the tag 'Tag Name' is not valid.

GLOSSARY

Α

Administrator

The person who uses the Configuration Manager to configure the contact center's resources and behaviors. An Administrator can be the Primary Administrator with unrestricted access to all the configuration objects or a Role Based Administrator with full or partical configuration rights.

Agent

Agents use the Agent Console to view and manage customer interactions. Virtual Contact Center supports Regular agents and Supervisor agents.

Agent Console

Virtual Contact Center's browser-based graphical user interface (GUI) used by Agents to manage customer interactions.

Application Programming Interface (API)

An application programming interface (API) defines the way an external program can request services from another program. See also, Virtual Contact Center Integrations.

Automatic Call Distribution (ACD)

ACD uses the number called and programmable call distribution logic to route incoming telephone calls to agents with the skills necessary to efficiently process the interaction.

Automatic Number Identification (ANI)

ANI identifies the origination number of a call offered to an Agent.

В

Billing Telephone Number (BTN)

For customers with multiple separate telephone numbers, the BTN consolidates those multiple numbers under a single phone number for billing purposes.

C

Call Treatment Choice

Refers to choices within a schedule which allows a time based selection of treatment choices when interactions enter a Virtual Contact Center tenant.

Caller Name (CNAM)

An enhanced Caller ID feature that displays both the number and name for an incoming voice call. To provide the value-added CNAM service, the carrier uses the originating caller's phone number to locate the caller's name in the carriers' names database, then transmits both the number and name to the CNAM-enabled customer.

Calling Line Identifier (CLI)

The intended display number of the calling party. In some network configurations, the CLI contains the calling party's Billing Telephone Number (BTN) or Charged Party Number (CPN) to be displayed instead.

Channel

Phone, email, or Web chat interactions arrive at a tenant on a channel. Virtual Contact Center uses skills based routing rules to place interaction requests into the appropriate Queue.

Collaborate

See Desktop Sharing.

Computer Telephony Integration (CTI)

CTI allows data associated with an incoming call or chat request to be presented to the responding agent in real-time. Virtual Contact Center CTI typically presents this data as a "screen pop".

Configuration Manager (CM)

The portion of the Virtual Contact Center application used by the contact center's administrator to define and configure the resources and operational behaviors of the center's tenant.

CRM API

The CRM API component of the Virtual Contact Center Integrations enables third-party processes to access the internal CRM data.

Custom Role

An administrator defined role with custom privileges is called a custom role. Any role other than the system defined Super User role is referred to as the custom role.

Customer Relationship Management (CRM)

The Virtual Contact Center CRM stores the contact center's customer, case, and follow-up data. Numerous third-party CRM products also exist. Two examples of third-party CRM applications are Salesforce and NetSuite.

D

Desktop Sharing

If authorized, an agent can use the Collaborate Desktop Sharing feature to view and control a customer's remote computer for purposes of providing assistance.

Direct Agent Access

In an IVR script, direct agent access permits a caller to reach an agent directly rather than through skills-based routing logic.

E

Email Script

A set of instructions that determine and direct the treatment and routing of an incoming email interaction.

G

Group

A group is a collection of agents created for management or reporting purposes. Groups can be functional (for example, Technical Support) or organizational (for example Supervisor Mary's Team). Each agent can be assigned to only one group, and cannot view or access information about other groups or group members.

Interactive Chat Response (ICR)

Interactive Chat Response specifies the message sent in response to a new chat interaction.

Interactive Voice Response (IVR)

IVR enables touch tones to guide customer interaction with the contact center. For example, an IVR script could direct a caller to Press 1 to be connected to Sales or Press 2 to be contacted to Support.

Intra Day Scheduler

Refers to the functionality of the tenant that allows creating multiple recurring choices within a day.

IVR Script

A set of instructions that prompts callers to choose options, then uses those choices to determine the optimal routing of an incoming phone interaction. See also, Interactive Voice Response.

O

Outbound Phone Codes

Outbound Phone codes offer a means to set a specific calling line identifier (caller ID) and optional outbound queue for outbound calls from your tenant. Outbound Phone Codes also may be used to track the purpose of an outbound call.

P

Primary Administrator

An adminstrator with unrestricted access to all the configuration objects in the Configuration Manager.

Q

Queue

A queue is an ordered collection of interactions waiting to be served by agents qualified to respond to these interactions. In addition to enabling the call center administrator to customize how incoming interactions are prioritized and routed, queues also ensure that interactions are never lost or discarded.

R

Reporting API

The Reporting API component of the Virtual Contact Center Integrations enables third-party processes to access statistical reporting data, and status information.

Responsible Organization (RespOrg)

The entity that controls and maintains a phone number. Telephone carriers are the most common type of RespOrg. 8x8's provisioning department processes the RespOrg orders required to direct your contact center's phone numbers to your Virtual Contact Center client

Role

A role defines a set of privileges to configure a Virtual Contact Center tenant. A role can be system defined or administrator defined.

Role Based Administrator

An administrator who assumes a role with full or partial configuration rights to the Virtual Contact Center.

Role Based Management

Role Based Management is the functionality of Virtual Contact Center that allows a tenant to distribute the configuration management functionality among multiple roles.

S

Screen Pop

The Screen Pop component of the Virtual Contact Center Integrations enables third-party CRMs to display information in the Agent Desktop. See also, Computer Telephony Integration.

Service Level Agreement (SLA)

A commitment to process some number of interactions within a specific period of time. Service Levels are generally defined as X percent of interactions answered within Y seconds measured over Z minutes. In the Configuration Manager, use a queue's SLA tab to generate an alert whenever interaction processing levels fall below a specified commitment level.

Skill

A qualification that associates a particular queue to an agent.

Skill Levels

For each agent, Virtual Contact Center uses high, medium, and low skill level identifiers to prioritize the distribution of interactions across the pool of available agents.

Status Codes

The Status Codes feature enables agents to supply supplemental information about why they changed their state. Status Codes provide enhanced workforce planning and management. For example, status codes could be created for different categories of breaks such as Break for Lunch, Break for Meeting, and so on.

Super User

The system defined role with unrestricted access to configure all administrative areas of the Configuration Manager. Primary Administrator inherits the Super User role automatically. Multiple administrators can be assigned to the Super User role. Privileges to the Super User role cannot be edited or revoked.

Supervisor

An agent with the supplementary privileges required to create FAQ categories and answers, monitor agent interactions in progress, listen to recordings of previous interactions, and create reports for the agents groups they supervise.

Support Center

In Configuration Manager, use the Support Center page to create customer-facing support Web page for your contact center, and specify the contact -- FAQs, desktop sharing, chat, email -- available from that page.

T

Tenant

Your unique and secure contact center instance running on a 8x8 platform. You use the Configuration Manager to create and configure all aspects of your contact center's resources and operational behaviors, including groups, incoming channels, queues, agents, routing scripts, skills, monitoring, recording, and reporting.

Transaction Code

The Transaction Codes feature enables agents to supply supplemental information about the reason for or outcome of an interaction. Transaction codes provide enhanced metrics or enable the calling line ID on an outbound call to be modified. For example, transaction codes could be created for different categories of sales activities such as Initial Contact, Prospect, Customer Satisfied, and so on.

Triggers

The Triggers component of the Virtual Contact Center Integrations invokes a remote third-party process whenever an agent creates, deletes, or edits internal CRM record.

V

Virtual Contact Center

A contact center that enables agents to be located in different geographical locations yet managed as if they were located in the same physical location.

Virtual Contact Center Integrations Suite

An optional package containing the CRM API, Screen Pop, Triggers, and Reporting API components.

Voice over IP (VoIP)

The use of the Internet Protocol (IP) to carry telephone calls.

W

Web Application Programming Interface (WAPI)

Deprecated term for the optional 8x8 Integrations package.

Weekly Schedule

Defines recurring day-of-week/time-of-day treatment choices. These are typically normal business hours. Schedules follow the defined weekly pattern unless superseded by date-specific Special Events.