



# ConcourseSuite CRM

## Sample API Usage

Matt Rajkowski

Revision 1.2

2011-07-11 8:00 AM EST



<b>Introduction</b>	<b>3</b>
<b>Setting up the CRM System</b>	<b>4</b>
<i>Steps for allowing a client to connect</i>	<b>4</b>
<i>Determine the CRM values to be used for associating the data to be added</i>	<b>4</b>
<b>Using the API</b>	<b>5</b>
<i>Protocol notes</i>	<b>5</b>
<i>Server URL</i>	<b>5</b>
<i>XML to save an Account w/Phone Number, Email Address and Street Address</i>	<b>5</b>
<i>Snippet when inserting General Contacts (without an Account)</i>	<b>6</b>
<i>Snippet when inserting Leads</i>	<b>7</b>
<i>Example Server Response</i>	<b>7</b>

# Introduction

ConcourseSuite includes a RESTful API which allows for various clients to communicate with the CRM using HTTP POST and XML.

For example, clients can capture leads or tickets from an existing web site and save them to ConcourseSuite. Most data in the CRM can be retrieved, inserted, updated and deleted.

This document assumes that the developer has knowledge about XML and using HTTP to POST data from their own applications, and that the developer has access to a ConcourseSuite CRM system.

## Online Resources

<http://www.concursive.com/show/concoursuite-support/wiki/Using+the+HTTP-XML+API>

<http://www.concursive.com/show/concoursuite-support/wiki/Tools+Package>

<http://www.concursive.com/show/concoursuite-support/wiki/XML+API+for+PHP>

# Setting up the CRM System

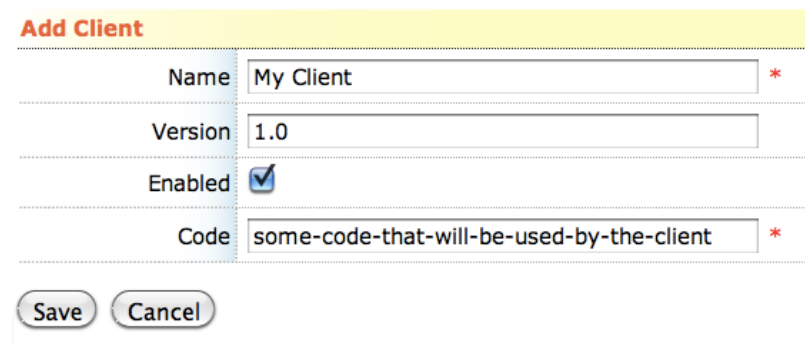
To enable the API, a CRM Administrator must login to the CRM and create an HTTP-XML API Client.

## Steps for allowing a client to connect

Navigate to:

Admin-> Configure System-> then choose “Edit” next to HTTP-XML API Client Manager

The page displays any clients that have been added. To add a client choose “Add Client” and fill out the form.



**Add Client**

Name	<input type="text" value="My Client"/>	*
Version	<input type="text" value="1.0"/>	
Enabled	<input checked="" type="checkbox"/>	
Code	<input type="text" value="some-code-that-will-be-used-by-the-client"/>	*

The code is the authentication value that will be used during the XML transaction.

*Upon saving, note the 'id' that the client belongs to.*

## Determine the CRM values to be used for associating the data to be added

Some research needs to be done in the CRM so that the data that is being added is correctly related to CRM users and lookup list values.

1. Determine the CRM user in which the recorded Account and related data will be associated with. Navigate to Admin-> Manage Users, then rollover or click on any user to inspect the URL. The url contains an ‘id’ parameter and value of the user.  
<https://127.0.0.1/Users.do?command=UserDetails&id=1>  
This id is used for all **enteredBy**, **modifiedBy**, and **owner** fields in the XML packet.
2. Confirm the relational id values for Phone Number Type, Email Type, and Address Type. By default, the value of “1” is used unless the CRM lookup list values have been modified. Navigate to Admin-> Configure Modules-> Accounts-> Lookup Lists, then view the HTML source of the page to see the IDs. These will be used in the XML packet.

# Using the API

The client application will be sending XML content to the CRM's API URL.

## Protocol notes

- Always use valid XML -- many applications can programmatically generate XML to ensure the output is valid. Concourse distributes a Java tools.jar which handles and simplifies the creation of XML and the posting of data to the API.
- Make sure to use "POST" for the request method
- Make sure to use the connection property: "Content-Type", "text/xml; charset="utf-8"

## Server URL

The client application will need to post the XML to the CRM's API URL:

<http://127.0.0.1/ProcessPacket.do>

## XML to save an Account w/Phone Number, Email Address and Street Address

Use the following XML, substituting the values between the CDATA brackets with the form's values. The authentication block must match the domain name of the server and the configured CRM client values. System Id 4 is a ruleset that the server uses and must be included as well. The transaction id is any arbitrary number and will be returned with the response.

```
<?xml version='1.0' encoding='UTF-8'?>
<app>
  <authentication>
    <id>127.0.0.1</id>
    <clientId>1</clientId>
    <code>PUT CLIENT CODE FROM CRM HERE</code>
    <systemId>4</systemId>
  </authentication>
  <transaction id='1'>
    <meta>
      <property>orgId</property>
      <property>name</property>
    </meta>
    <account action='insert' shareKey='true'>
      <name><![CDATA[Practice Name]]></name>
      <owner>${default}</owner>
      <enteredBy>${default}</enteredBy>
      <modifiedBy>${default}</modifiedBy>
      <dunsType><![CDATA[Client Type]]></dunsType>
```

```

    <dunsNumber><![CDATA[NPI]]></dunsNumber>
    <sicText><![CDATA[UPIN]]></sicText>
    <sicDescription><![CDATA[CampCode:Keyword]]></sicDescription>
</account>
<organizationPhoneNumber action='insert'>
  <orgId>${C}{account.id}</orgId>
  <number><![CDATA[Phone Number]]></number>
  <type>1</type>
  <enteredBy>${U}{default}</enteredBy>
  <modifiedBy>${U}{default}</modifiedBy>
</organizationPhoneNumber>
<organizationEmailAddress action='insert'>
  <orgId>${C}{account.id}</orgId>
  <email><![CDATA[Email]]></email>
  <type>1</type>
  <enteredBy>${U}{default}</enteredBy>
  <modifiedBy>${U}{default}</modifiedBy>
</organizationEmailAddress>
<organizationAddress action='insert'>
  <orgId>${C}{account.id}</orgId>
  <streetAddressLine1><![CDATA[Address]]></streetAddressLine1>
  <city><![CDATA[City]]></city>
  <state><![CDATA[State]]></state>
  <zip><![CDATA[Zip]]></zip>
  <country>UNITED STATES</country>
  <type>1</type>
  <enteredBy>${U}{default}</enteredBy>
  <modifiedBy>${U}{default}</modifiedBy>
</organizationAddress>
<contact action='insert' shareKey='true'>
  <orgId>${C}{account.id}</orgId>
  <primaryContact>true</primaryContact>
  <accessType>${AT}{ACCOUNT_CONTACT_PUBLIC}</accessType>
  <nameFirst><![CDATA[First Name]]></nameFirst>
  <nameLast><![CDATA[Last Name]]></nameLast>
  <owner>${U}{default}</owner>
  <enteredBy>${U}{default}</enteredBy>
  <modifiedBy>${U}{default}</modifiedBy>
</contact>
</transaction>
</app>

```

## Snippet when inserting General Contacts (without an Account)

```

<...>
  <contact action='insert' shareKey='true'>
    <accessType>${AT}{GENERAL_CONTACT_PUBLIC}</accessType>
    <nameFirst><![CDATA[First Name]]></nameFirst>
    <nameLast><![CDATA[Last Name]]></nameLast>
    <owner>${U}{default}</owner>
    <enteredBy>${U}{default}</enteredBy>
    <modifiedBy>${U}{default}</modifiedBy>
  </contact>
</...>

```

```
</contact>
<...>
```

## Snippet when inserting Leads

```
<...>
  <contact action='insert' shareKey='true'>
    <isLead>true</isLead>
    <leadStatus>1</leadStatus>
    <accessType>$AT{GENERAL_CONTACT_PUBLIC}</accessType>
    <nameFirst><![CDATA[First Name]]></nameFirst>
    <nameLast><![CDATA[Last Name]]></nameLast>
    <company><![CDATA[Some company name]]></company>
    <enteredBy>$U{default}</enteredBy>
    <modifiedBy>$U{default}</modifiedBy>
  </contact>
  <contactEmailAddress action='insert'>
    <contactId>$C{contact.id}</contactId>
    <email><![CDATA[Email]]></email>
    <type>1</type>
    <enteredBy>$U{default}</enteredBy>
    <modifiedBy>$U{default}</modifiedBy>
  </contactEmailAddress>
  <contactPhoneNumber action='insert'>
    <contactId>$C{contact.id}</contactId>
    <number><![CDATA[888-555-1212]]></number>
    <type>1</type>
    <enteredBy>$U{default}</enteredBy>
    <modifiedBy>$U{default}</modifiedBy>
  </contactPhoneNumber>
<...>
```

## Example Server Response

The server will respond with XML indicating a SUCCESS or FAILURE. If a failure, then an error message is also returned... Tomcat's web logs will have the exact error message from the server's perspective.

## Success

```
<?xml version="1.0" encoding="UTF-8"?>
<aspcfs>
  <response id="1">
    <status>0</status>
    <errorText/>
    <recordSet count="1" name="account">
      <record action="processed" name="">
        <orgId>638</orgId>
        <name>Practice Name</name>
      </record>
    </recordSet>
  </response>
</aspcfs>
```

## Failure

```
<?xml version="1.0" encoding="UTF-8"?>
<aspcfs>
  <response id="1">
    <status>1</status>
    <errorText>Transaction failed</errorText>
  </response>
</aspcfs>
```