

# Alianza Web Service Introduction

---

**Revision History:**

#	By	Date	Notes
1	M. Ryan Werner		Initial
2	M. Ryan Werner	11/23/09	Modified with corrected URLs

---

#

Introduction .....	2
Web Service Interface .....	2
Alianza Web Services.....	2
Web Service WSDL Definition .....	3
Web Service Security.....	3
ClientCode.....	3
SecurityKey .....	3
AuthToken.....	3
Web Service Input Parameters .....	3
Web Service Return Values .....	4

---

## Introduction

The Alianza Web Service APIs are a collection of SOAP/XML-based web methods that can be leveraged by Service Providers to interact with their customers' accounts. The Web Methods are grouped into various Web Services in a logical manner, either by action category, or by the object the methods operate on. A listing of these services and their relative endpoints can be found below.

## Web Service Interface

Applications will interact with the Alianza Web Services API by passing SOAP-encapsulated XML objects via HTTP. There are currently two environments in which the web services function. These two environments are separate and distinct from one another, so that what happens in one will not affect the other. One is a development environment where our service provider clients will do their development work with the web services API. The other environment is the Production (Live) environment.

Development and testing of an application will be done primarily by calling the web service methods in the lab environment, and then when the application is deployed, or perhaps when more final testing is occurring, the application should be pointed at the Production (Live) environment.

To access any of the Alianza Web Services, an application is pointed at the following URLs:

Development Environment:

<http://services.beta.alianza.com:1701/4.0/>

Production (Live) Environment:

<https://services.alianza.com:1701/4.0/>

## Alianza Web Services

Below is a list of the Alianza Web Services, and the relative endpoint that applies to each service.

Service Name	Contents / Purpose	Relative Endpoint
Account	Functions related directly to accounts	/Account.asmx
Billing	Functions related to billing; primarily retrieval of billing transactions	/Billing.asmx
CallHandling	Functions revolving around call handing for extensions or phone numbers	/CallHandling.asmx
CallingPlan2	Functions relating to the changing and listing of calling plans	/CallingPlan2.asmx
Device	Functions relating to device management	/Device.asmx
DialingRestriction	Functions relating to use and implementation of Dialing Restrictions	/DialingRestriction.asmx
E911	Functions relating to E911 service in the US (availability per contract specifications)	/E911.asmx
Extension	Functions related to management of extensions	/Extension.asmx
Geography	Functions related to geography (country, state, city code lists)	/Geography.asmx
Misc	Various miscellaneous functions	/Misc.asmx
MonthlyFee	Functions to support the activation or deactivation of specific monthly fees (MRC)	/MonthlyFee.asmx
Notification	Functions supporting the use of the End-User (on-login) notification system	/Notification.asmx
PhoneNumber	Functions to support the management of phone numbers	/PhoneNumber.asmx
PlatformConfig	Functions relating to platform configuration	/PlatformConfig.asmx
Search	Various search functions	/Search.asmx
SIPTrunk	Functions relating to management of SIP Trunks	/SIPTrunk.asmx
Site	Functions relating to the management of sites	/Site.asmx
Softphone	Functions relating to softphone (GoFone) management	/Softphone.asmx
Staging	Functions duplicating most functionality present in the Staging Manager in ATAC	/Staging.asmx

Transaction	Functions related to creating a specific transaction and applying it against a specific account	/Transaction.asmx
User	Functions related to the management of end-user account users	/User.asmx

To access the methods that make up the Account Web Service, the endpoint used by the application would be:

`https://servi ces. al ianza. com: 1701/Account. asmx`

### Web Service WSDL Definition

It is possible to obtain a WSDL definition of a specific Alianza Web Service by appending a “?wsdl” command to the end of the endpoint for the service, as seen here:

`https://servi ces. al ianza. com: 1701/Account. asmx?wsdl`

### Web Service Security

The web service security is implemented through the use of the three input parameters below. All three of these parameters are passed into each web method.

#### ClientCode

The ClientCode is 6 characters in length and there is only one ClientCode associated with each Alianza client.

#### SecurityKey

The SecurityKey is 25 characters long, and is specific to the ClientCode. When you pass the SecurityKey to the web service you will need to encode it using Base64.

#### AuthToken

AuthToken is both an input *and* an output parameter. The AuthToken will be provided back to the application as a return parameter from each method. On the first call to a web method, the application can pass in a zero-length string for the AuthToken.

If a value is passed in, the AuthToken is checked by the web service against the ClientCode, SecurityKey and time, and if found to be valid there is no further validation performed. This feature can slightly increase the performance of the API calls.

If the AuthToken is found to be invalid, or expired, the ClientCode and SecurityKey are validated by the web service against the Alianza system and a new AuthToken is issued.

### Web Service Input Parameters

The current API Web Methods expect the values of the input parameters to be passed as either a list of “primitive” objects -- Integers, Strings, etc; or as attributes of a “complex” XML object.

A list of “primitive” objects is shown here in an example input to the Account/Account\_List method:

```

<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <Account_List xmlns="http://www.alianza.com/webservices/">
      <ClientCode>string</ClientCode>
      <b64SecurityKey>string</b64SecurityKey>
      <AuthToken>string</AuthToken>
      <AccountTypeFK>int</AccountTypeFK>
      <LanguageFK>int</LanguageFK>
      <OperatorUserFK>int</OperatorUserFK>
      <FilterString>string</FilterString>
      <SortByNameOrNbr>int</SortByNameOrNbr>
    </Account_List>
  </soap:Body>
</soap:Envelope>

```

A set of parameters as attributes of a "complex" XML object is demonstrated by this call to the Account/Account\_Update\_CustomFields method:

```

<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <Account_Update_CustomFields xmlns="http://www.alianza.com/webservices/">
      <objIn>
        <ClientCode>string</ClientCode>
        <b64SecurityKey>string</b64SecurityKey>
        <AuthToken>string</AuthToken>
        <AccountFK>int</AccountFK>
        <Custom01>string</Custom01>
        <Custom02>string</Custom02>
        <Custom03>string</Custom03>
        <Custom04>string</Custom04>
        <Custom05>string</Custom05>
        <TicketFK>int</TicketFK>
        <OperatorUserFK>int</OperatorUserFK>
      </objIn>
    </Account_Update_CustomFields>
  </soap:Body>
</soap:Envelope>

```

## Web Service Return Values

All Web Service methods return their outputs as attributes of a "complex" object, as shown in this example output from the Account/Account\_Update\_CustomFields method:

```

<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <Account_List_ByCustomFieldsResponse
xmlns="http://www.alianza.com/webservices/">
      <Account_List_ByCustomFieldsResult>
        <ReturnValue>int</ReturnValue>
        <AuthToken>string</AuthToken>
        <OutputList>
          <Account_List_ByCustomFields_Output_ListItem>
            <ClientAccountNbr>string</ClientAccountNbr>
            <AccountNbr>string</AccountNbr>
            <AccountPK>int</AccountPK>
          </Account_List_ByCustomFields_Output_ListItem>
          <Account_List_ByCustomFields_Output_ListItem>
            <ClientAccountNbr>string</ClientAccountNbr>
            <AccountNbr>string</AccountNbr>
            <AccountPK>int</AccountPK>
          </Account_List_ByCustomFields_Output_ListItem>
        </OutputList>
      </Account_List_ByCustomFieldsResult>
    </Account_List_ByCustomFieldsResponse>
  </soap:Body>
</soap:Envelope>

```