**Ciirus XML Web Services**

**Version 12.017**

**Contents**

[General Information](#h.gjdgxs)

[What’s new](#h.30j0zll)

[Version 12.017](#h.1fob9te)

[Web Service URL](#h.3znysh7)

[Summary of Functions available within the XML](#h.2et92p0)

[Passing Dates](#h.tyjcwt)

[Function Descriptions](#h.3dy6vkm)

[The GetProperties Function](#h.1t3h5sf)

[The FilterOptions structure](#h.4d34og8)

[The SearchOptions structure](#h.17dp8vu)

[The GetCommunities Function](#h.3rdcrjn)

[The GetDescriptions Function](#h.26in1rg)

[The GetImageList Function](#h.lnxbz9)

[The GetManagementCompanyList Function](#h.35nkun2)

[The GetPropertyClasses Function](#h.1ksv4uv)

[The GetPropertyRates Function](#h.44sinio)

[The GetPropertyTypes Function](#h.2jxsxqh)

[The GetReservations Function](#h.z337ya)

[The IsPropertyAvailable Function](#h.3j2qqm3)

[The MakeBooking Function](#h.1y810tw)

[The sBookingDetails structure](#h.4i7ojhp)

[The sGuests structure](#h.2xcytpi)

[Example Code](#h.1ci93xb)

[Adding the web service to your project](#h.3whwml4)

[Referencing the web service from your code](#h.2bn6wsx)

[Getting a property list](#h.qsh70q)

[Getting the Reservations](#h.3as4poj)

[Placing a Booking](#h.1pxezwc)

# General Information

## 

## What’s new

### Version 12.017

New function: **MakeBooking** – you can now place a booking via XML (contact Ciirus for pricing details).

Updated: **GetPropertyRates** – this now applies agent daily rate mark-up and multiplication factor (if applicable) to returned rates.

Updated: **GetProperties** – now returns new element CurrencyCode (e.g. USD or EUR).

New Option: An XML user can now be flagged to return all properties on a property search even if the property manager has not enabled some properties to show on their website. To set this, contact Ciirus who will update your user account.

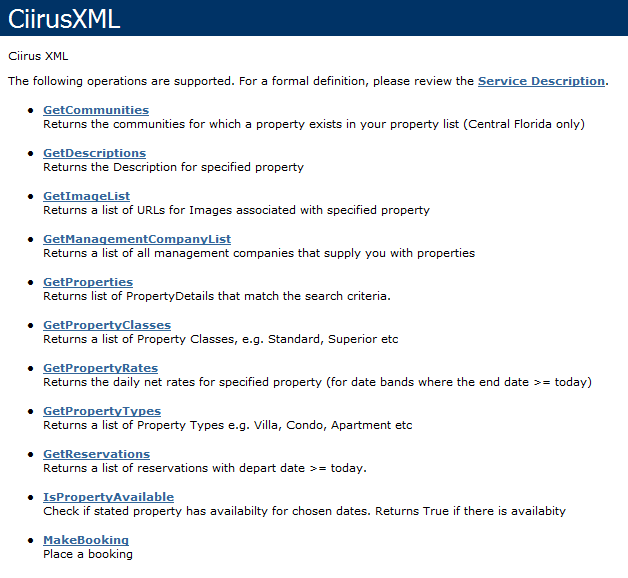
Bug Fix: Fixed an issue whereby some users could not authenticate to some web functions.

## Web Service URL

The web service URL is as below

<http://xml.ciirus.com/CiirusXML.12.017.asmx>

## Summary of Functions available within the XML feed



To view the service description, go to the URL above, and then select any of the functions displayed to see details of the parameters and return values.

The web services return standard SOAP responses. The examples in the documentation below are based on .NET but the functions can be used from any platform that supports the use of web services, such as C#, VB.NET, PHP and others.

Each function must be passed your API username and password. The username and password will be provided to you by Ciirus, but you can request these be changed at any time if necessary.

The XML web services are under on-going development, and can be enhanced as necessary if you have a need for additional functionality.

## Passing Dates

For functions that require dates as a parameter (such as quoting and checking availability) you must pass the dates as a string. The format of the dates must be dd MMM yyyy

The day part can be 1 or 2 digits (1 – 31)

The month part must be either the first 3 letters of the month or the full month name (e.g. Jan or January is fine, but Sept will throw an error)

The year must be the 4 digit year

# Function Descriptions

## The GetProperties Function

Use this function to get a list of properties that are available to you. This will include properties listed under your own Ciirus account as well as any properties assigned to you by a supplier.

You can pass various filter options to the function. If you pass a date range then only properties that are available for your specified dates will be returned. If you pass the start or end date as a NULL or Empty string then all properties that match your other filter criteria will be returned regardless of availability.

**Pass the following parameters**

APIUsername

APIPassword

ArriveDate (dd MMM yyyy) – Pass empty string if you do not want to filter by availability

DepartDate (dd MMM yyyy) – Pass empty string if you do not want to filter by availability

FilterOptions – Details of this structure below

SearchOptions – Details of this structure below

### The FilterOptions structure

|  |  |  |
| --- | --- | --- |
| Element | Data Type | Comments |
| ManagementCompanyID | Integer | Pass 0 for properties from all management companies. To retrieve only properties from a specific management company, enter their ID. Use the ‘GetManagementCompanyList’ function to get the list of available management companies and their ID. |
| CommunityID | Integer | This is currently for Central Florida Only. Pass 0 for all communities, else pass the ID of the community to filter. Use the ‘GetCommunities’ function to get the list of communities in which you have properties available. |
| PropertyID | Integer | Pass 0 for all properties or a specific ID to only receive details for 1 property |
| PropertyType | Integer | Pass 0 for all types or a type ID for a specific type only. Use the ‘GetPropertyTypes’ function to get the list of available types |
| HasPool | Integer | Pass 0 for properties without a pool, 1 for properties with a pool or any other integer for all properties |
| HasSpa | Integer | Pass 0 for properties without a spa, 1 for properties with a spa or any other integer for all properties |
| PrivacyFence | Integer | Pass 0 for properties without a privacy fence, 1 for properties with a privacy fence or any other integer for all properties |
| CommunalGym | Integer | Pass 0 for properties without a communal gym, 1 for properties with a communal gym or any other integer for all properties |
| HasGamesRoom | Integer | Pass 0 for properties without a games room, 1 for properties with a games room or any other integer for all properties |
| IsGasFree | Boolean | Pass True to return only properties with no gas. Pass False for all properties |
| Sleeps | Integer | Pass 0 for all properties, or >0 for properties that sleep at least the specified amount of guests |
| Bedrooms | Integer | Pass 0 for all properties, or >0 for properties with the specified number of bedrooms |
| PropertyClass | Integer | Pass 0 for all classes or a class ID for a specific class only. Use the ‘GetPropertyClasses’ function to get the list of available classes |
| ConservationView | Integer | Pass 0 for properties without a conservation view, 1 for properties with a conservation view or any other integer for all properties |
| WaterView | Integer | Pass 0 for properties without a water view, 1 for properties with a water view or any other integer for all properties |
| LakeView | Integer | Pass 0 for properties without a lake view, 1 for properties with a lake view or any other integer for all properties |
| WiFi | Integer | Pass 0 for properties without Wi-Fi, 1 for properties with Wi-Fi or any other integer for all properties |
| PetsAllowed | Integer | Pass 0 for properties that do not allow pets, 1 for properties that do allow pets or any other integer for all properties |
| OnGolfCourse | Integer | Pass 0 for properties not on a golf course, 1 for properties on a golf course or any other integer for all properties |
| SouthFacingPool | Integer | Pass 0 for properties without a south facing pool, 1 for properties with a south facing pool or any other integer for all properties |

### The SearchOptions structure

|  |  |  |
| --- | --- | --- |
| Element | Data Type | Comments |
| ReturnTopX | Integer | Pass 0 to return all results, or >0 to return only that amount of results |
| ReturnFullDetails | Boolean | Pass False to return fewer details in the search results. This will make the results download to you faster. Pass True to return all property details |
| ReturnQuote | Boolean | Pass False if you do not need a quote returned for each property. Pass True to return a quote for the specified dates. ArriveDate and DepartDate must be entered if you wish to return a quote.  Note: The time it takes for a query to run is much increased when you elect to return a quote. Do not select this option unless you specifically need to see the quoted price against each property. |
| IncludePoolHeatInQuote | Boolean | This option only applies if you also set ‘ReturnQuote’ to True. Setting this option to True includes the cost of pool heating in the calculated quote. |

Response

The following structure is returned as an array

<PropertyDetails>

<PropertyID>int</PropertyID>

<ManagementCompanyName>string</ManagementCompanyName>

<MainImageURL>string</MainImageURL>

<MCPropertyName>string</MCPropertyName> *(this is the name used by the management company)*

<WebsitePropertyName>string</WebsitePropertyName> *(this is the name as it should appear on your website)*

<DescriptionSetID>int</DescriptionSetID>

<Community>string</Community>

<Bedrooms>int</Bedrooms>

<Sleeps>int</Sleeps>

<Bathrooms>decimal</Bathrooms>

<XCO>decimal</XCO> *(this is the X map co-ordinate)*

<YCO>decimal</YCO> *(this is the Y map co-ordinate)*

<Bullet1>string</Bullet1> *(5 main bullet points)*

<Bullet2>string</Bullet2>

<Bullet3>string</Bullet3>

<Bullet4>string</Bullet4>

<Bullet5>string</Bullet5>

<CurrencySymbol>string</CurrencySymbol>

<CurrencyCode>string</CurrencyCode>

<HasPool>boolean</HasPool>

<HasSpa>boolean</HasSpa>

<GamesRoom>boolean</GamesRoom>

<PropertyType>string</PropertyType>

<PropertyClass>string</PropertyClass>

<HostMCUserID>int</HostMCUserID> *(this is the ID of the management company the property belongs to)*

<QuoteExcludingTax>decimal</QuoteExcludingTax> *(If an error creating quote or less than the minimum allowed night’s stay then this will be zero, so check before displaying this quote to guest)*

<QuoteIncludingTax>decimal</QuoteIncludingTax> *(If an error creating quote or less than the minimum allowed night’s stay then this will be zero, so check before displaying this quote to guest)*

<LessThanMinimumNightsStay>boolean</LessThanMinimumNightsStay> *(True if the requested dates are less than the minimum number of nights permitted)*

<MinimumNightsStay>int</MinimumNightsStay> *(Minimum nights a guest is allowed to book for this property. 0=no minimum)*

<CommunalPool>boolean</CommunalPool>

<CommunalGym>boolean</CommunalGym>

<ConservationView>boolean</ConservationView>

<WaterView>boolean</WaterView>

<GolfView>boolean</GolfView>

<GolfIncluded>boolean</GolfIncluded>

<WiFi>boolean</WiFi>

<Internet>boolean</Internet>

<WiredInternetAccess>boolean</WiredInternetAccess>

<PetsAllowed>boolean</PetsAllowed>

<Stroller>boolean</Stroller>

<Crib>boolean</Crib>

<HighChair>boolean</HighChair>

<AirCon>boolean</AirCon>

<ElectricFireplace>boolean</ElectricFireplace>

<GasFireplace>boolean</GasFireplace>

<WoodBurningFireplace>boolean</WoodBurningFireplace>

<FreeCalls>boolean</FreeCalls>

<FreeSolarHeatedPool>boolean</FreeSolarHeatedPool>

<PacknPlay>boolean</PacknPlay>

<PrivacyFence>boolean</PrivacyFence>

<SouthFacingPool>boolean</SouthFacingPool>

<BBQ>boolean</BBQ>

<AirHockeyTable>boolean</AirHockeyTable>

<BigScreenTV>boolean</BigScreenTV>

<CDPlayer>boolean</CDPlayer>

<DVDPlayer>boolean</DVDPlayer>

<FoosballTable>boolean</FoosballTable>

<Grill>boolean</Grill>

<HairDryer>boolean</HairDryer>

<TVInEveryBedroom>boolean</TVInEveryBedroom>

<VCR>boolean</VCR>

<IndoorJacuzzi>boolean</IndoorJacuzzi>

<IndoorHotTub>boolean</IndoorHotTub>

<OutdoorHotTub>boolean</OutdoorHotTub>

<PavedParking>boolean</PavedParking>

<PoolAccess>boolean</PoolAccess>

<RockingChairs>boolean</RockingChairs>

<VideoGames>boolean</VideoGames>

<Fishing>boolean</Fishing>

<BedroomConfiguration>string</BedroomConfiguration>

<Address1>string</Address1>

<Address2>string</Address2>

<City>string</City>

<County>string</County>

<State>string</State>

<Zip>string</Zip>

<Country>string</Country>

<Telephone>string</Telephone>

</PropertyDetails>

## The GetCommunities Function

Use this function to get a list of communities that apply to the properties that are available to you.

**Pass the following parameters**

APIUsername

APIPassword

**Response**

The following structure is returned as an array

<Communities>

<CommunityID>int</CommunityID>

<CommunityName>string</CommunityName>

</Communities>

## The GetDescriptions Function

Use this function to get the description for the specified property. The description that is returned is in HTML format and does not contain <html>, <head> or <body> tags

**Pass the following parameters**

APIUsername

APIPassword

PropertyID

**Response**

A single string containing the description is returned

## The GetImageList Function

Use this function to get a string array of URLs to images for the specified property.

**Pass the following parameters**

APIUsername (string)

APIPassword (string)

PropertyID (integer)

**Response**

A string array, each containing a URL to an image

## The GetManagementCompanyList Function

Use this function to get a list of the management companies that supply properties to you. Your own company will always be returned in this list, so even if you have no suppliers then at least 1 record will be returned with your User ID and company name.

**Pass the following parameters**

APIUsername (string)

APIPassword (string)

**Response**

The following structure is returned as an array

<ManagementCompanies>

<MCUserID>int</MCUserID>

<MCCompanyName>string</MCCompanyName>

</ManagementCompanies>

## The GetPropertyClasses Function

Use this function to get a list of the property classes available, along with their ID’s (this ID is used to filter the search results in the GetProperties function)

**Pass the following parameters**

APIUsername (string)

APIPassword (string)

**Response**

The following structure is returned as an array

<PropertyClasses>

<ID>int</ID>

<PropertyClass>string</PropertyClass>

</PropertyClasses>

## The GetPropertyRates Function

Use this function to get a list of the daily rates assigned to the specified property. The rate will be returned inclusive of any defined mark-ups (if you are an agent). Note, prior to version 12.017, the rate was returned with no mark-ups.

Only rate bands where the end date is greater than today’s date are returned.

Rates are returned exclusive of tax

**Pass the following parameters**

APIUsername (string)

APIPassword (string)

PropertyID (integer)

**Response**

The following structure is returned as an array

<Rate>

<FromDate>dateTime</FromDate>

<ToDate>dateTime</ToDate>

<MinNightsStay>int</MinNightsStay>

<DailyRate>decimal</DailyRate>

</Rate>

## The GetPropertyTypes Function

Use this function to get a list of the property types available, along with their ID’s (this ID is used to filter the search results in the GetProperties function)

**Pass the following parameters**

APIUsername (string)

APIPassword (string)

**Response**

The following structure is returned as an array

<PropertyTypes>

<ID>int</ID>

<PropertyType>string</PropertyType>

</PropertyTypes>

## The GetReservations Function

Use this function to get a list of the reservations in the specified property.

Only reservations where the departure date is greater than or equal to today’s date are returned.

**Pass the following parameters**

APIUsername (string)

APIPassword (string)

PropertyID (integer)

**Response**

The following structure is returned as an array

<Reservations>

<ArrivalDate>dateTime</ArrivalDate>

<DepartureDate>dateTime</DepartureDate>

</Reservations>

## The IsPropertyAvailable Function

Use this function to do a real-time check if the specified property is available.

**Pass the following parameters**

APIUsername (string)

APIPassword (string)

PropertyID (integer)

ArrivalDate (string, dd MMM yyyy)

DepartureDate (string, dd MMM yyyy)

**Response**

True = Property is available

False = Property is not available

## The MakeBooking Function

Use this function to place a booking. The booking is created with the rate calculated based on the parameters you pass.

**Pass the following parameters**

APIUsername (string)

APIPassword (string)

BookingDetails (sBookingDetails)

### The sBookingDetails structure

|  |  |  |
| --- | --- | --- |
| **Element** | **Data Type** | **Comments** |
| ArrivalDate | String | Booking arrival date in format ‘dd MMM yyyy’ |
| DepartureDate | string | Booking departure date in format ‘dd MMM yyyy’ |
| PropertyID | Integer | The Ciirus Property ID |
| GuestName | String | The lead guest name |
| GuestEmailAddress | String | The guest’s email address |
| GuestTelephone | String | The guest’s telephone number |
| GuestAddress | String | The guest’s address |
| GuestList | sGuests | Each guest’s name and age (if applicable). See table below for the structure |
| PoolHeatRequired | Boolean | If True then then this is added and included in the calculated price |

### The sGuests structure

|  |  |  |
| --- | --- | --- |
| **Element** | **Data Type** | **Comments** |
| Name | String | Guest Name |
| Age | Integer | Age of guest if applicable (else 0) |

**The following structure is returned (MakeBookingResult)**

|  |  |  |
| --- | --- | --- |
| **Element** | **Data Type** | **Comments** |
| BookingPlaced | Boolean | True if successful. False if failed. If failed then the error will be in the ‘ErrorMessage’ element |
| ErrorMessage | String | Empty if booking was successful, else will contain the error message |
| BookingID | Integer | If the booking was successful then this will contain the Ciirus Booking ID |
| TotalAmountIncludingTax | Decimal | This is the total booking amount inclusive of tax |

# Example Code

The following examples are based on .NET although the web services can be consumed by any language that supports XML Web Services

## Adding the web service to your project

In Visual Studio, right click your project in the Solution Explorer and ‘Add Service Reference’

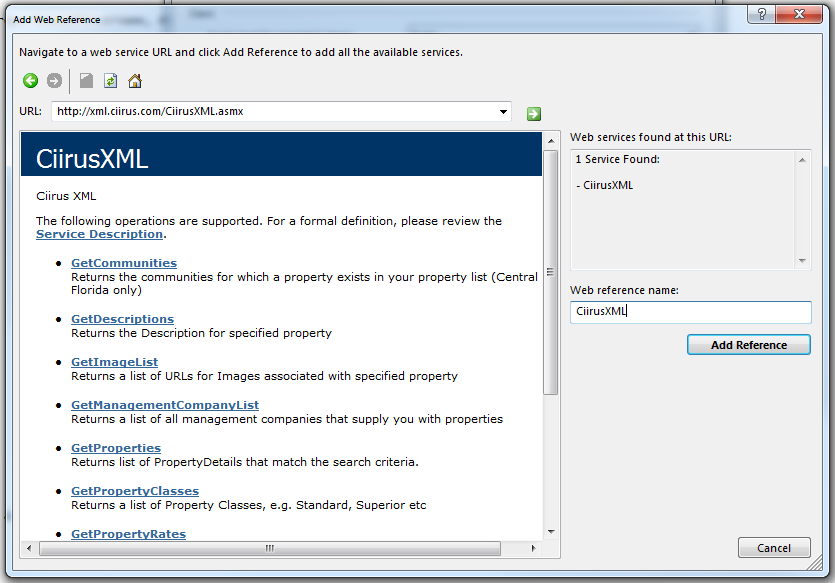
If you are using Visual Studio 2010 or higher, click the ‘Advance’ button then ‘Add Web Reference’

Where prompted for the URL, enter the path to the Ciirus XML web service file such as

<http://xml.ciirus.com/CiirusXML.12.017.asmx>

Click the Green -> icon to the right and you should see the list of available functions appear in the window below.

Enter a web reference name where prompted (e.g. CiirusXML) then click the ‘Add Reference’ button to complete the process.



## Referencing the web service from your code

Create variable to create your web service such as

Dim CXML As New CiirusXML.CiirusXML

You can now reference CXML any time you wish to call functions from the Ciirus XML web service.

## Getting a property list

**Step 1 -** Create reference to the Ciirus XML

Dim CXML As New CiirusXML.CiirusXML

**Step 2 -** Configure your filter options

Dim FO As New CiirusXML.strucSearchFilterOptions

FO.Bedrooms = 4 ‘4 bedroom properties only

FO.CommunityID = 0 ‘Any community

FO.HasGamesRoom = 1 ‘Must have games room

FO.HasPool = 1 ‘Must have pool

FO.HasSpa = -1 ‘Either without without Spa

FO.IsGasFree = False

FO.PropertyClass = 0 ‘Any property class

FO.PropertyType = 0 ‘Any property type

FO.Sleeps = 8 ‘Property must be able to sleep at least 8 guests

**Step 3 -** Configure your search options

Dim SO As New CiirusXML.strucSearchOptions

SO.ReturnTopX = 0 ‘Return all results

SO.ReturnQuote = False ‘Do not calculate quotes for each property returned

SO.ReturnFullDetails = True ‘Return all details available for each property

**Step 4** – Call the web function

Dim APIUsername As String = “11006” ‘replace with your assigned username

Dim APIPassword As String = “11006” ‘replace with your assigned password

Dim ArrivalDate As String = "1 June 2013"

Dim DepartureDate As String = "8 June 2013"

Dim PropertyList() As CiirusXML.PropertyDetails = CXML.GetProperties(APIUsername, APIPassword, ArrivalDate, DepartureDate, FO, SO)

**Step 5** – Accessing the results

The variable “PropertyList()” now contains an array of type “PropertyDetails”. You can access the individual records as below

Get the number of records:

Dim NumRecords As Integer = PropertyList.Length

Get the Property Name of the first record

Dim PN As String = PropertyList(0).WebsitePropertyName

Get all properties in results

For Each R As CiirusXML.PropertyDetails In PropertyList

Dim PID As Integer = R.PropertyID

Dim PN As String = R.WebsitePropertyName

Dim MC As String = R.ManagementCompanyName

'Process / display the results

'

'

Next

## Getting the Reservations

To get a list of all future reservations for a specified property use the following function:

Dim Reservations() As CiirusXML.Reservations = CXML.GetReservations(APIUsername, APIPassword, 32480)

This returns an array of type “Reservations” that can be accessed as follows

For Each R As CiirusXML.Reservations In Reservations

Dim AD As Date = R.ArrivalDate

Dim DD As Date = R.DepartureDate

Next

## Placing a Booking

Dim BD As New CiirusXML.sBookingDetails

BD.ArrivalDate = “1 Sep 2012”

BD.DepartureDate = “8 Sep 2012”

BD.GuestAddress = "1234 High Street, Kissimmee, Florida, 34554, USA"

BD.GuestEmailAddress = “test@test.com”

BD.GuestName = “Guest Name”

BD.GuestTelephone = "111 222 3333"

BD.PoolHeatRequired = False

BD.PropertyID = 12345

Dim GL(1) As CiirusXML.sGuests

GL(0) = New CiirusXML.sGuests

GL(0).Name = "Guest 1 Name"

GL(0).Age = 0

GL(1) = New CiirusXML.sGuests

GL(1).Name = "Child Guest 2"

GL(1).Age = 12

BD.GuestList = GL

Dim BR As CiirusXML.sBookingResult = CXML.MakeBooking(APIUsername, APIPassword, BD)

If BR.BookingPlaced Then

Dim BID as integer = BR.BookingID

Dim TotalPrice as Decimal = BR.TotalAmountIncludingTax

Else

Dim ER as string = BR.ErrorMessage

End If