Companies Office   
BusinessData Web Services  
Searching – Contract Model

Version 1.7

# Document control

## Version history

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Version | Author | Description of change |
| 13/10/2011 | 0.1 | Gary Dodd | Initial Draft |
| 13/12/2011 | 1.0 | Gary Dodd | Incorporated changes from Tim Lum |
| 15/03/2012 | 1.1 | Gary Dodd | Link to xsd corrected. Reference to SOAP headers removed. Further example for string to sign included to assist with signature generation. |
| 11/05/2012 | 1.2 | Gary Dodd | Insert Troubleshoot assistance for String to sign. |
| 02/11/2012 | 1.3 | Haydn Flower | Updated URL to businessdata.govt.nz |
| 16/11/2012 | 1.4 | Haydn Flower | Updated Maori character Hex Values |
| 16/11/2012 | 1.5 | Sonni Nielsen | Update Entity Detail example and XSD section |
| 9/12/2013 | 1.6 | Haydn Flower | Added some code examples |
| 28/02/2014 | 1.7 | Haydn Flower | V2 Services released |

Contents

1. Document control 2

Version history 2

2. Overview 4

Service Description 4

Entity Search 4

Role Search 4

Entity Details 5

3. BusinessData Authentication 6

Introduction 6

HMAC (Hashed Message Authentication Code) 6

Signature Generation 6

Further Example for a String to sign: 7

Issues with String to sign – Troubleshooting: 8

4. Request/Response Standards 9

RESTful Request Structure 9

5. Service Operations 10

Search Services 10

Companies Entity Search 10

Notes 10

Companies Role Search 11

Notes 11

Companies Entity Details Service 12

6. Examples 13

Entity Search 13

Role Search 14

Entity Detail 15

7. XML Schema 18

8. Appendix 1 33

List of Permitted Characters for Searching 33

List of Diacritics 34

List of all Entity Status Values 34

List of All Entity Body Type Values 35

List of All Country Code Values 35

# Overview

## Service Description

Contained in this contract model is a set of web service operations which allow clients to search the Companies Office database for specific entity information.

The following services are available within this contract model:

1. Entity Search
2. Role Search
3. Entity Details

## Entity Search

The Entity Search service searches the Companies Office database on current or historic company details, using company name, company number or one or two keywords within the company name.

It reproduces the functionality of the Companies Office website search facility.

Results returned via this service provide summary information about the company, similar to the search results page on the Companies Office website.

|  |  |
| --- | --- |
| Service Property | Value |
| Name | CompaniesEntitySearch |
| Description | Company Entity Search |
| Version | 1.0 and 2.0 |
| Application/Registry | Companies |
| URL | http://www.businessdata.govt.nz/data/app/ws/rest/companies/entity/search/v1.0/ |

Technical information regarding this service is explained in the Service Operations Section of this contract model.

## Role Search

The Role Search service allows clients to search the Companies Office database for details on Directors, Shareholders (current and historic) and also search the Banned Director and Manager database.

This search reproduces the functionality of the Companies Office website seach facility for Directors and Shareholders.

Results returned via this service provide details of the directorships and shareholdings for individuals and shareholdings for companies and other entities.

|  |  |
| --- | --- |
| Service Property | Value |
| Name | CompaniesRoleSearch |
| Description | Company Role Search |
| Version | 1.0 |
| Application/Registry | Companies |
| URL | http://www.businessdata.govt.nz/data/app/ws/rest/companies/role/search/v1.0/ |

Technical information regarding this service is explained in the Service Operations Section of this contract model.

## Entity Details

The Entity Details service returns all information held for a specific company. This includes:

* Core company information
  + This includes Company name, number, nzbn, status, registration date, etc.
* Address details (including historic address data)
* Director details (including historic directorships)
* Shareholder details (including historic shareholdings)
* Documents registered (including a link direct to that document)
* Non-legislative data
  + This includes Trading name, website address, BIC code, NZSX code and other optional information provided by the company.

|  |  |
| --- | --- |
| Service Property | Value |
| Name | CompaniesEntityDetails |
| Description | Company Entity Details |
| Version | 1.0 and 2.0 |
| Application/Registry | Companies |
| URL | http://www.businessdata.govt.nz/data/app/ws/rest/companies/entity/v1.0/ |

Technical information regarding this service is explained in the Service Operations Section of this contract model.

# BusinessData Authentication

## Introduction

BusinessData is the authentication service layer through which all business services are exposed. Each business service transaction is first authenticated in BusinessData before forwarding the transaction to the appropriate business service for processing.

Each RESTful service call contains authentication data in the request header. The service specific data is embedded in the RESTful URL. This is then forwarded to Companies to get the required information which is returned to the client as an XML string or JSON string.

## HMAC (Hashed Message Authentication Code)

HMAC is the authentication mechanism used by BusinessData. A shared **Secret Authentication Key** is used by the client to create a signature based upon predefined message properties. Upon receiving the signature with the request the service will recalculate the signature to authenticate the user’s identity.

Each request should provide the signature and the data required as part of the source.

Two pieces of information are required to generate a signature.

* Authentication Key ID: This is your unique identifier to verify that you are a BusinessData user. This will not change.
* Secret Authentication Key: Used to generate a signature from the “String to Sign”. This Secret Authentication Key is never transmitted in a message and should never be passed to a client machine. A client’s secret key is available on the BusinessData website and a facility is in place to generate a new Secret Authentication Key if required.

## Signature Generation

The “String to Sign” is the final string which is signed using a SHA256 digest to create the signature. It is built up from the following attributes of an HTTP request.

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute | **Description** | **Required/Optional** | **Supplied As** |
| HTTP Method | HTTP method used for the request. Note that all BusinessData services only support GET requests. | Required |  |
| Host | Standard HTTP header. | Required |  |
| Request URI | URI of service being requested. | Required |  |
| Timestamp | Time that the signature was generated. | Required | Request header in the standard HTTP format as defined in RFC1123,eg: “Sun, 15 May 2011 19:12:01 GMT” |
| Authentication Key ID | Identifies the BusinessData client | Required | As part of the Request “Authorisation” header. |
| Accept | Standard HTTP Accept header. | Required | Either “application/xml” or “application/json” |
| Context | A client identifier for the transaction. If supplied, this must be unique to each service usage. (ie: user and service combination). | Optional | Request header. |

The “String to Sign” is built up from these values separated by a new line character (‘\n’). The construction of the String to Sign follows the template shown below:

<HTTP Method>\n<Host>\n<Request URI>\n<Timestamp>\n<Authentication Key ID>\n<Accept>\n<Context>\n

#### Example Source String

GET

www.businessdata.govt.nz

/data/app/ws/rest/companies/entity/search/v1.0/carran%20miller

Sun, 15 May 2011 19:12:01 GMT

Application/xml

T4WZUVoufkaGfsrK

A Base64 encoded SHA256 hash is then applied to this string using your Secret Authentication Key (not the Authentication Key ID) to produce the signature. An example signature may look like the following:

wFtZpfdDAh9kul6ZEnlvvmRYdWmvqjLxagNUcpMYGTM=

This signature must be added to the request as a header in the form <Authentication Key ID>:<Signature>:

Authorization:"T4WZUVoufkaGfsrK:4yDg0FE8nzqhY80K+5w1sHXVO+6UBXeMG4irvoUx6jI="

If no context element is supplied the signature will only be valid for 20 seconds after its generation time. The timestamp user for generation on the server must match that to be supplied in the request header. It is recommended that the server be synchronised with a NTP server.

## Further Example for a String to sign:

This example 'string to sign' comprises the following component strings concatenated together in the order they are listed (note: there is also an optional "Context" field which is not included in this example). **Every** component string must end with a new-line character (including the last one).

(1) "GET"  
(2) URL domain of the yet-to-be-sent http request  
(3) URL path of the yet-to-be-sent http request  
(4) the value of the 'date' field in the yet-to-be-sent http request header  
(5) the 'access key' which identifies the BusinessData client  
(6) the value of the 'accept' field in the yet-to-be-sent http request header  
  
The 'string to sign' for our example is :  
  
GET**¶**www.dev.business.govt.nz**¶**/data/app/ws/rest/companies/entity/v1.0/1277262**¶**Wed, 22 Feb 2012 02:23:00 GMT**¶**T4WZUVoufkaGfsrK**¶**application/xml,text/html,application/xhtml+xml,application/xml;q=0.9,\*/\*;q=0.8**¶**  
  
(where **¶** denotes a new-line character) (note: there are 5 space characters in the above 'string to sign', and all are in the date field)  
  
  
Given the 'string to sign' shown above and a **Secret Authentication Key** of 86MVgkk3fdiLlzPCXaXDGCn22DSqEVFJ8GNzKGED then the 'signature' (or 'authorization string') computed and then supplied in the http request header must be : QSYqcxEF6pT0Ds5cZRI6Tz708CjLomF0dZ0LDzX1IFw=  
  
To aid in developing your client application code, it is suggested that this example 'string to sign' and 'secret authentication key' should be run through that code to ensure the computed 'signature' exactly matches the value indicated above.

## Issues with String to sign – Troubleshooting:

The following is a summary of the issues we have had so far with regards to third party access to the new Companies web services:

These can be summarised into two areas:

1. putting together the 'string to sign' correctly, and
2. using the correct algorithm for generating the 'signature' (from the 'string to sign')

Area 2 should be sorted out if you use the example (see above) and make sure your code gets the same signature as that shown.

**The String to sign needs to be EXACTLY the same as the headers in your Request that you send,**

**otherwise you will end up with a 401 response.**

Area 1 can be broken down further:

1. forgetting to put a single {newline} character at the **end** of the entire string
2. forgetting to put a forward-slash at the **beginning** of the URL component
3. not using **exactly** the same date string for both the header and 'string to sign'. That is; the request header "date" field must be the same value used in the 'string to sign'
4. accidentally changing the format of the date string when computing the 'signature'. For example, your program may store the date value in a "date" variable. When this is converted to a string (to become part of the 'string to sign') that action may inadvertently convert the date into local (NZT) time. "Wed, 22 Feb 2012 02:23:00 GMT" may change into "Thur, 23 Feb 2012 04:23:00 NZT" for example, and even though these might represent the same date, this string would be different from the one used by our server to validate the request.
5. If the order of your parameters are different

Request /data/app/ws/rest/companies/entity/search/v1.0?page=12&q=stuff&size=3

Signature /data/app/ws/rest/companies/entity/search/v1.0?q=stuff&page=12&size=3

1. If you use Microsoft .Net, use the HMACSHA256 class and pass the secret key to the constructor
2. Header elements in a different order

Request:

GET

eat.businessdata.govt.nz

/data/app/ws/rest/companies/entity/search/v1.0/bayf

Wed, 28 Nov 2012 03:34:29 GMT

application/json

BKudm5FlGM3Hjtl5

Signature:

GET

eat.businessdata.govt.nz

/data/app/ws/rest/companies/entity/search/v1.0/bayf

Wed, 28 Nov 2012 03:34:29 GMT

BKudm5FlGM3Hjtl5

application/json

If you have checked all of the above and still have issues, we suggest you resend your request to our EAT server and provide us with two bits of info:

1. The exact time you submitted (to makes things easier for us - the more recent, the better). We need this to find your request in the log file.
2. The exact 'string to sign' that your code used to compute the 'signature'. This must be the actual value used for the request, so you must somehow capture this as part of your code which sends the request

|  |  |
| --- | --- |
| Restful Service | Service URI Example |
| Companies Entity Search | http://www.businessdata.govt.nz/data/app/ws/rest/companies/entity/search/v1.0/carran%20millar |
| Companies Entity Detail | http://www.businessdata.govt.nz/data/app/ws/rest/companies/entity/v1.0/1000000 |
| Companies Role Search | http://www.businessdata.govt.nz/data/app/ws/rest/companies/role/search/v1.0/hone |

# Request/Response Standards

## RESTful Request Structure

All the services described in this document are RESTful web services. Each transaction must contain the BusinessData client’s authentication data and the input data required by the business service.

* Authentication Data – The client’s authentication data must be supplied in the request header.
* Input Data – The service specific input data is embedded in the request URL.

# Service Operations

## Search Services

There are two RESTful search services.

URL paths for these services are flexible, allowing parameters to be supplied either as components to the path or as query string parameters.

## Companies Entity Search

|  |  |
| --- | --- |
| Service | Entity Search |
| **HTTP Method** | GET |
| **Return Element** | entitySearchResults |
| **Return Type** | entitySearchResults |

Base URL: [http://www.businessdata.govt.nz/data/app/ws/rest/companies/entity/search/v1.0](http://www.business.govt.nz/data/app/ws/rest/companies/entity/search/v1.0)

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Description | Required/Optional | Valid patterns |
| q | Search parameter | Required | {baseURL}?q={query\_string}  {baseURL}/{query\_string} |
| Size | Page size  Default = 50 | Optional | {baseURL}?q={query\_string}&page={page}&size={size}  {baseURL}/page/{page}/size/{size}/{query\_string} |
| Page | Page number  Default = 0  (0 indexed) | Optional | {baseURL}?q={query\_string}&page={page}  {baseURL}/page/{page}/{query\_string} |

Base URL: <http://www.businessdata.govt.nz/data/app/ws/rest/companies/entity/search/v2.0>

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Description | Required/Optional | Valid patterns |
| q | Search parameter | Required | {baseURL}?q={query\_string}  {baseURL}/{query\_string} |
| Size | Page size  Default = 50 | Optional | {baseURL}?q={query\_string}&page={page}&size={size}  {baseURL}/page/{page}/size/{size}/{query\_string} |
| Page | Page number  Default = 0  (0 indexed) | Optional | {baseURL}?q={query\_string}&page={page}  {baseURL}/page/{page}/{query\_string} |

## Notes

1. If non-alphanumeric characters are to be supplied as part of a query, it is recommended to use the parameterised versions over the url path equivalent. If searching for a term containing an “&”, “/”, “%” these must be percent-encoded as “%26”, “%2F”, “%25”. Further details on percent encoding can be found on [Wikipedia](http://en.wikipedia.org/wiki/Percent_encoding) or as defined in [RCF 3986](http://tools.ietf.org/html/rfc3986).
2. A set of allowed special characters is defined in the Appendix.
3. All company body types and company status values are listed in the Appendix.

## Companies Role Search

|  |  |
| --- | --- |
| Service | Role Search |
| **HTTP Method** | GET |
| **Return Element** | roleSearchResults |
| **Return Type** | roleSearchResults |

Base URL: [http://www.businessdata.govt.nz/data/app/ws/rest/companies/role/search/v1.0](http://www.business.govt.nz/data/app/ws/rest/companies/role/search/v1.0)

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Description | Required/Optional | Valid patterns |
| q | Search parameter | Required | {baseURL}?q={query\_string}  {baseURL}/{query\_string} |
| Size | Page size  Default = 50 | Optional | {baseURL}?q={query\_string}&page={page}&size={size}  {baseURL}/page/{page}/size/{size}/{query\_string} |
| Page | Page number  Default = 0  (0 indexed) | Optional | {baseURL}?q={query\_string}&page={page}  {baseURL}/page/{page}/{query\_string} |
| Type | Role type  Default = ALL | Optional | {baseURL}?q={query\_string}&page={page}&size={size}&type={type}  {baseURL}/page/{page}/size/{size}/type/{type}/{query\_string}  **Search types:**   * **ALL** = Searches both Director and Shareholder roles.\* * **DIR** = Director role only * **SHR** = Shareholder role only * **DDIR** = Disqualified/Banned Director   \* The “All” search excludes disqualified/banned directors. |

## Notes

1. A minimum of 2 characters is required to complete a search

## Companies Entity Details Service

|  |  |
| --- | --- |
| Service | Entity Detail |
| **HTTP Method** | GET |
| **Return Element** | entityDetail |
| **Return Type** | entityDetail |

Base URL: [http://www.businessdata.govt.nz/data/app/ws/rest/companies/entity/v1.0](http://www.business.govt.nz/data/app/ws/rest/companies/entity/v1.0)

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Description | Required/Optional | Valid patterns |
| Company number | * Company number to retrieve. * Must be numeric, up to 7 digits. * Must exist on the Companies Register. | Required | {baseURL}/{company\_number} |

Base URL: <http://www.businessdata.govt.nz/data/app/ws/rest/companies/entity/v2.0>

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Description | Required/Optional | Valid patterns |
| Company number | * Company or NZBN number to retrieve. * Must be numeric. * Must exist on the Companies Register. | Required | {baseURL}/{company\_number} |

# V2 Services

With the release of the NZBN a new version of the Entity Search and Entity Details services have been developed to show the nzbn element in the result sets

The new v2.0 URLs:

business.govt.nz/data/app/ws/rest/companies/entity/v2.0/{CompanyID}

business.govt.nz/data/app/ws/rest/companies/entity/search/v2.0/{CompanyID}

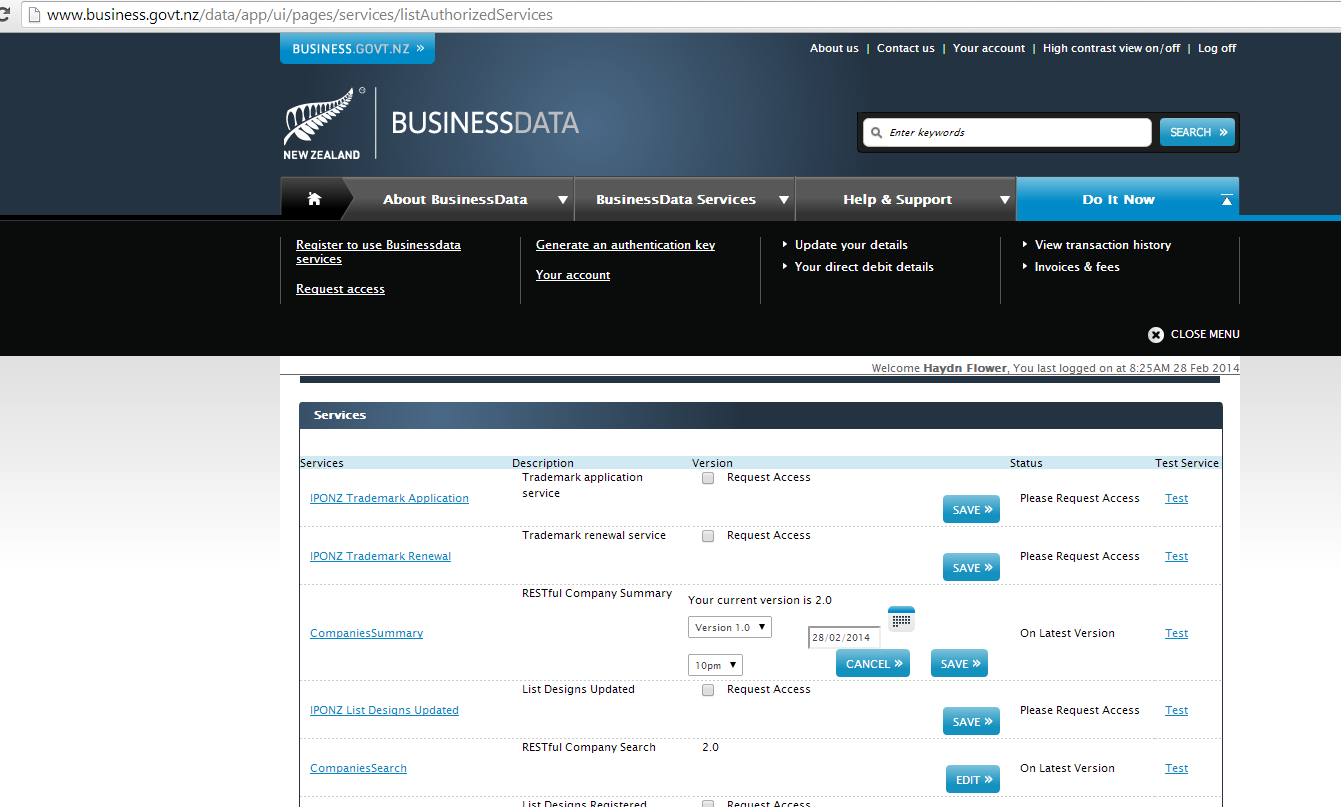
Each follow the1\_2.xsd found at

<http://www.business.govt.nz/data/app/ws/schemas/companies/rest/v1.2/companiesServices_v1_2.xsd>

we have renamed the Single Business Identifier element to NZBN and added it to the shareholders information and Entity Search Results.

In both the v1.0 and v2.0 web services you can use a NZBN as a search string as well as the Company ID.

To start using the V2 services, you need to make a request on the following ‘Request access’ page and Edit the verison details.



If you are getting “Invalid Service Request“ errors then you are probably making requests to V1 when you are signed up to V2 or vice versa.

You can’t make requests to multiple versions of the services with the same Access Key.

# Examples

## Entity Search

### Request

GET http://www.eat.businessdata.govt.nz/data/app/ws/rest/companies/entity/search/v1.0/page/0/size/2/agile%20test HTTP/1.1

Host: www.eat.businessdata.govt.nz

User-Agent: Mozilla/5.0 (X11; U; Linux x86\_64; en-US; rv:1.9.1.16) Gecko/20110429 Iceweasel/3.5.16 (like Firefox/3.5.16)

Accept: application/xml,text/html,application/xhtml+xml,application/xml;q=0.9,\*/\*;q=0.8

Date: Wed, 25 May 2011 21:59:24 GMT

Context: 1306360764876

Authorization: YRXqtBKWSa5H4qdc:SP3zY++yN5wO/qyyx6Va+0oK8Oek37GRXHtQGzljtJA=

### Response

HTTP/1.1 200 OK

Content-Type: application/xml

Content-Language: en

Server: This server is configured to not send version information (N;ecid=141048705695760,0)

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>

<entitySearchResults>

<currentPage>0</currentPage>

<pageSize>2</pageSize>

<companies>

<company xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="entitySearchResult">

<companyName>AGILE TESTING SOLUTIONS LIMITED</companyName>

<entityType>

<code>U</code>

<description>Limited Liability Company</description>

</entityType>

<companyNumber>1850999</companyNumber>

<incorporationDate>2006-08-15T00:00:00.000+12:00</incorporationDate>

<registeredOfficeAddress>

<addressLines>5 Wimbledon Way</addressLines>

<addressLines>Remuera</addressLines>

<addressLines>Auckland</addressLines>

<postCode>1050</postCode>

<countryCode>NZ</countryCode>

</registeredOfficeAddress>

<status>

<code>60</code>

<description>In Liq</description>

</status>

</company>

<company xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="entitySearchResult">

<companyName>AGILITY LIMITED</companyName>

<entityType>

<code>U</code>

<description>Limited Liability Company</description>

</entityType>

<companyNumber>1938365</companyNumber>

<incorporationDate>2007-05-22T00:00:00.000+12:00</incorporationDate>

<registeredOfficeAddress>

<addressLines>9 Richard Pearse Drive, Mangare</addressLines>

<addressLines>Auckland</addressLines>

<countryCode>NZ</countryCode>

</registeredOfficeAddress>

<status>

<code>50</code>

<description>Registered</description>

</status>

</company>

</companies>

<totalResults>544</totalResults>

</entitySearchResults>

## Role Search

### Request

GET http://www.eat.businessdata.govt.nz

/data/app/ws/rest/companies/role/search/v1.0/page/0/size/2/frank HTTP/1.1

Host: www.eat.businessdata.govt.nz

User-Agent: Mozilla/5.0 (X11; U; Linux x86\_64; en-US; rv:1.9.1.16) Gecko/20110429 Iceweasel/3.5.16 (like Firefox/3.5.16)

Accept: application/xml,text/html,application/xhtml+xml,application/xml;q=0.9,\*/\*;q=0.8

Accept-Language: en-us,en;q=0.5

Accept-Encoding: gzip,deflate

Date: Wed, 25 May 2011 22:18:05 GMT

Context: 1306361885712

Authorization: YRXqtBKWSa5H4qdc:7y0GihsCnxeNYlcbmqaPVF7KiA0B9OiUI+Fo7LsjDfw=

### Response

HTTP/1.1 200 OK

Content-Type: application/xml

Content-Language: en

Server: This server is configured to not send version information (N;ecid=141048705695760,0)

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>

<roleSearchResults>

<currentPage>0</currentPage>

<pageSize>2</pageSize>

<roles>

<roles xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="entityOrganisationShareholder">

<physicalAddress>

<addressLines>Starkenburgstrasse 1</addressLines>

<addressLines>64546 Moerfelden-Walldorf</addressLines>

<addressLines>Germany</addressLines>

</physicalAddress>

<status>active</status>

<name>FRANK GMBH</name>

<shareholdings>

<associatedCompanyName>PPS - FRANK N.Z. LIMITED</associatedCompanyName>

<associatedCompanyNumber>1620976</associatedCompanyNumber>

<jointlyHeld>false</jointlyHeld>

<numberOfShares>483885</numberOfShares>

</shareholdings>

</roles>

<roles xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="entityOrganisationShareholder">

<physicalAddress>

<addressLines>C/ Bdo Spicers Gisborne</addressLines>

<addressLines>1 Peel Street</addressLines>

<addressLines>Gisborne</addressLines>

</physicalAddress>

<status>active</status>

<name>FRANK BLACK LIMITED</name>

<shareholdings>

<associatedCompanyName>SQUASH HARVEST LIMITED</associatedCompanyName>

<associatedCompanyNumber>928345</associatedCompanyNumber>

<jointlyHeld>true</jointlyHeld>

<numberOfShares>1500</numberOfShares>

<sharePercentage>50</sharePercentage>

</shareholdings>

</roles>

</roles>

<totalResults>11683</totalResults>

</roleSearchResults>

## Entity Detail

### Request

GET http://www.eat.businessdata.govt.nz/data/app/ws/rest/companies/entity/v1.0/1277262 HTTP/1.1

Host: www.eat.businessdata.govt.nz

User-Agent: Mozilla/5.0 (X11; U; Linux x86\_64; en-US; rv:1.9.1.16) Gecko/20110429 Iceweasel/3.5.16 (like Firefox/3.5.16)

Accept: application/xml,text/html,application/xhtml+xml,application/xml;q=0.9,\*/\*;q=0.8

Date: Wed, 25 May 2011 22:28:30 GMT

Context: 1306362510012

Authorization: YRXqtBKWSa5H4qdc:qz7NDEHY4abv6T4XiOyVX9mdEdvS+ypAJUoy5Y81520=

### Response

HTTP/1.1 200 OK

Content-Type: application/xml

Content-Language: en

Server: This server is configured to not send version information (N;ecid=141048705695760,0)

<entityDetail>

<companyName>NEW NEW LIMITED</companyName>

<companyNumber>1277262</companyNumber>

<companyUri>[http://www.eat.business.govt.nz/companies/app/ui/pages/companies/1277262</companyUri](http://www.eat.business.govt.nz/companies/app/ui/pages/companies/1277262%3c/companyUri)>

<incorporationDate>2003-02-20+13:00</incorporationDate>

<entityType>

<code>LTD</code><description>NZ Limited Company</description>

</entityType>

<status><code>50</code>

<description>Registered</description>

</status>

<annualReturnFilingMonth>2</annualReturnFilingMonth>

<countryOfOrigin><code>NZ</code><description>New Zealand</description></countryOfOrigin>

<inRemoval>false</inRemoval>

<hasConstitutionFiled>true</hasConstitutionFiled>

<reRegistered>false</reRegistered>

<totalNumberOfShares>100</totalNumberOfShares>

<extensiveShareholding>false</extensiveShareholding>

<registeredOfficeAddress>

<addressLines>17, 4f Crownlynn Place</addressLines>

<addressLines>New Lynn</addressLines>

<addressLines>Auckland</addressLines>

<countryCode>NZ</countryCode>

</registeredOfficeAddress>

<addressForService>

<addressLines>17, 4f Crownlynn Place</addressLines>

<addressLines>New Lynn</addressLines>

<addressLines>Auckland</addressLines>

<countryCode>NZ</countryCode></addressForService>

<previousCompanyStatuses>

<code>50</code>

<description>Registered</description><dateChanged>2003-02-20+13:00</dateChanged>

</previousCompanyStatuses>

<previousCompanyStatuses>

<code>80</code>

<description>Struck Off</description>

<dateChanged>2005-06-08+12:00</dateChanged>

</previousCompanyStatuses><previousCompanyStatuses><code>50</code><description>Registered</description><dateChanged>2012-09-24+12:00</dateChanged>

</previousCompanyStatuses>

<addressForRecords/>

<addressForShareRegister/>

<personsAuthorisedToAcceptService/>

<historicAddresses>

<registeredOfficeAddresses><addressLines>Unit 22, 167 Whitney Street</addressLines><addressLines>Blockhouse Bay</addressLines><addressLines>Auckland</addressLines>

<effectiveFrom>2003-02-20+13:00</effectiveFrom>

<effectiveTo>2004-03-18+13:00</effectiveTo>

</registeredOfficeAddresses>

<addressForService>

<addressLines>Unit 22, 167 Whitney Street</addressLines><addressLines>Blockhouse Bay</addressLines><addressLines>Auckland</addressLines>

<effectiveFrom>2003-02-20+13:00</effectiveFrom>

<effectiveTo>2004-03-18+13:00</effectiveTo>

</addressForService></historicAddresses>

<activeDirectors><director><appointmentDate>2003-02-20+13:00</appointmentDate>

<physicalAddress>

<addressLines>17, 4f, Crownlynn Place</addressLines><addressLines>New Lynn</addressLines>

<addressLines>Auckland</addressLines></physicalAddress>

<firstName>Lei</firstName>

<lastName>NIU</lastName></director>

</activeDirectors>

<inactiveDirectors/>

<shareParcels>

<shareParcel>

<shareholders xsi:type="individualRole"><physicalAddress><addressLines>17, 4f, Crownlynn Place</addressLines>

<addressLines>New Lynn</addressLines>

<addressLines>Auckland</addressLines>

<countryCode>NZ</countryCode>

</physicalAddress>

<firstName>Lei</firstName>

<lastName>NIU</lastName>

</shareholders>

<numberOfShares>100</numberOfShares></shareParcel>

</shareParcels>

<documents><filingRequirement><archived>false</archived><attachments><attachment><uri>http://www.eat.business.govt.nz/companies/app/service/services/documents/99766E4CC22C69444A34FFECFD30CA4A</uri></attachment></attachments><filingCode><code>RESREG</code><description>Company Restoration</description></filingCode><registrationDate>2012-09-24+12:00</registrationDate><url>http://www.eat.business.govt.nz/companies/app/ui/pages/companies/1277262/13862579/entityFilingRequirement</url></filingRequirement><filingRequirement><archived>false</archived><attachments/><filingCode><code>OAR</code><description>Online Annual Return</description></filingCode><registrationDate>2004-03-11+13:00</registrationDate><url>http://www.eat.business.govt.nz/companies/app/ui/pages/companies/1277262/3210777/entityFilingRequirement</url></filingRequirement><filingRequirement><archived>false</archived><attachments/><filingCode><code>OSH</code><description>Particulars of Shareholding</description></filingCode><registrationDate>2004-03-11+13:00</registrationDate><url>http://www.eat.business.govt.nz/companies/app/ui/pages/companies/1277262/3210752/entityFilingRequirement</url></filingRequirement><filingRequirement><archived>false</archived><attachments/><filingCode><code>OPD</code><description>Online Particulars of Directors</description></filingCode><registrationDate>2004-03-11+13:00</registrationDate><url>http://www.eat.business.govt.nz/companies/app/ui/pages/companies/1277262/3210599/entityFilingRequirement</url></filingRequirement><filingRequirement><archived>false</archived><attachments/><filingCode><code>OCA</code><description>Online Particulars Of Company Address</description></filingCode><registrationDate>2004-03-11+13:00</registrationDate><url>http://www.eat.business.govt.nz/companies/app/ui/pages/companies/1277262/3210111/entityFilingRequirement</url></filingRequirement><filingRequirement><archived>false</archived><attachments><attachment><uri>http://www.eat.business.govt.nz/companies/app/service/services/documents/527E66E0728FE1C0C60F558946C273A2</uri></attachment></attachments><filingCode><code>CSTSHR</code><description>Consent Of Shareholder</description></filingCode><registrationDate>2003-02-20+13:00</registrationDate><url>http://www.eat.business.govt.nz/companies/app/ui/pages/companies/1277262/12744222/entityFilingRequirement</url></filingRequirement><filingRequirement><archived>false</archived><attachments><attachment><uri>http://www.eat.business.govt.nz/companies/app/service/services/documents/CC95C49B77BB4BD3C19AF262945BE3FA</uri></attachment></attachments><filingCode><code>CSTOFF</code><description>Consent Of Director</description></filingCode><registrationDate>2003-02-20+13:00</registrationDate><url>http://www.eat.business.govt.nz/companies/app/ui/pages/companies/1277262/12744221/entityFilingRequirement</url></filingRequirement><filingRequirement><archived>false</archived><attachments><attachment><uri>http://www.eat.business.govt.nz/companies/app/service/services/documents/3CD50491D93DF8C2C03FB2FB41D1D017</uri></attachment></attachments><filingCode><code>CONO</code><description>Adoption/Amendment Of Constitution</description></filingCode><registrationDate>2003-02-19+13:00</registrationDate><url>http://www.eat.business.govt.nz/companies/app/ui/pages/companies/1277262/12743111/entityFilingRequirement</url></filingRequirement>

<filingRequirement><archived>false</archived><attachments/><filingCode><code>NRN</code><description>Application To Incorporate A Company</description></filingCode><registrationDate>2003-02-19+13:00</registrationDate>

<url>http://www.eat.business.govt.nz/companies/app/ui/pages/companies/1277262/12743097/entityFilingRequirement</url>

</filingRequirement></documents>

<additionalCompanyInformation><bicCode/><premiseAddress/>

</additionalCompanyInformation>

</entityDetail>

# XML Schema

The following XSD files contain the schema definitions for the Companies RESTful services defined in this document.

V1: <http://www.business.govt.nz/data/app/ws/schemas/companies/rest/v1.1/companiesServices_v1_1.xsd>

V2: <http://www.business.govt.nz/data/app/ws/schemas/companies/rest/v1.2/companiesServices_v1_2.xsd>

The structure for a JSON response is also based off this schema.

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>

<xs:schema version="1.0" xmlns:xs="http://www.w3.org/2001/XMLSchema">

<xs:element name="entityDetail" type="entityDetail"/>

<xs:element name="entityRequest" type="entityRequest"/>

<xs:element name="entitySearchResults" type="entitySearchResults"/>

<xs:element name="roleSearchResults" type="roleSearchResults"/>

<xs:complexType name="reportList">

<xs:sequence>

<xs:element name="report" type="report" minOccurs="0" maxOccurs="unbounded"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="report">

<xs:sequence>

<xs:element name="date" type="xs:date" minOccurs="0"/>

<xs:element name="filed" type="xs:boolean"/>

<xs:element name="name" type="xs:string" minOccurs="0"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="appointeeList">

<xs:sequence>

<xs:element name="appointee" type="appointee" minOccurs="0" maxOccurs="unbounded"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="appointee">

<xs:sequence>

<xs:element name="address" type="address" minOccurs="0"/>

<xs:element name="appointeeName" type="xs:string" minOccurs="0"/>

<xs:element name="appointmentDate" type="xs:date" minOccurs="0"/>

<xs:element name="emailAddress" type="xs:string" minOccurs="0"/>

<xs:element name="firstName" type="xs:string" minOccurs="0"/>

<xs:element name="lastName" type="xs:string" minOccurs="0"/>

<xs:element name="middleName" type="xs:string" minOccurs="0"/>

<xs:element name="name" type="xs:string" minOccurs="0"/>

<xs:element name="organisationName" type="xs:string" minOccurs="0"/>

<xs:element name="phoneNumber" type="xs:string" minOccurs="0"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="address">

<xs:sequence>

<xs:element name="careOf" type="xs:string" minOccurs="0"/>

<xs:element name="addressLines" type="xs:string" nillable="true" minOccurs="0" maxOccurs="unbounded"/>

<xs:element name="postCode" type="xs:string" minOccurs="0"/>

<xs:element name="countryCode" type="xs:string" minOccurs="0"/>

<xs:element name="countryName" type="xs:string" minOccurs="0"/>

<xs:element name="pafId" type="xs:string" minOccurs="0"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="externalAdministration">

<xs:sequence>

<xs:element name="appointees" type="appointeeList" minOccurs="0"/>

<xs:element name="description" type="xs:string" minOccurs="0"/>

<xs:element name="reports" type="reportList" minOccurs="0"/>

<xs:element name="status" type="xs:string" minOccurs="0"/>

<xs:element name="type" type="xs:string" minOccurs="0"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="entityRequest">

<xs:sequence>

<xs:element name="companyNumber" type="xs:string" minOccurs="0"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="director">

<xs:complexContent>

<xs:extension base="individualRole">

<xs:sequence>

<xs:element name="associatedCompanyName" type="xs:string" minOccurs="0"/>

<xs:element name="associatedCompanyNumber" type="xs:string" minOccurs="0"/>

</xs:sequence>

</xs:extension>

</xs:complexContent>

</xs:complexType>

<xs:complexType name="individualRole">

<xs:complexContent>

<xs:extension base="role">

<xs:sequence>

<xs:element name="firstName" type="xs:string" minOccurs="0"/>

<xs:element name="lastName" type="xs:string" minOccurs="0"/>

<xs:element name="middleName" type="xs:string" minOccurs="0"/>

</xs:sequence>

</xs:extension>

</xs:complexContent>

</xs:complexType>

<xs:complexType name="role" abstract="true">

<xs:sequence>

<xs:element name="appointmentDate" type="xs:date" minOccurs="0"/>

<xs:element name="physicalAddress" type="address" minOccurs="0"/>

<xs:element name="resignationDate" type="xs:date" minOccurs="0"/>

<xs:element name="secretaryResignationDate" type="xs:date" minOccurs="0"/>

<xs:element name="status" type="xs:string" minOccurs="0"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="entityOrganisationRole">

<xs:complexContent>

<xs:extension base="role">

<xs:sequence>

<xs:element name="name" type="xs:string" minOccurs="0"/>

<xs:element name="number" type="xs:string" minOccurs="0"/>

</xs:sequence>

</xs:extension>

</xs:complexContent>

</xs:complexType>

<xs:complexType name="entityOrganisationShareholder">

<xs:complexContent>

<xs:extension base="entityOrganisationRole">

<xs:sequence>

<xs:element name="shareholdings" type="shareHolding" nillable="true" minOccurs="0" maxOccurs="unbounded"/>

</xs:sequence>

</xs:extension>

</xs:complexContent>

</xs:complexType>

<xs:complexType name="shareHolding">

<xs:sequence>

<xs:element name="associatedCompanyName" type="xs:string" minOccurs="0"/>

<xs:element name="associatedCompanyNumber" type="xs:string" minOccurs="0"/>

<xs:element name="jointlyHeld" type="xs:boolean"/>

<xs:element name="numberOfShares" type="xs:long" minOccurs="0"/>

<xs:element name="sharePercentage" type="xs:string" minOccurs="0"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="individualShareholder">

<xs:complexContent>

<xs:extension base="individualRole">

<xs:sequence>

<xs:element name="shareholdings" type="shareHolding" nillable="true" minOccurs="0" maxOccurs="unbounded"/>

</xs:sequence>

</xs:extension>

</xs:complexContent>

</xs:complexType>

<xs:complexType name="directorShareholder">

<xs:complexContent>

<xs:extension base="individualShareholder">

<xs:sequence/>

</xs:extension>

</xs:complexContent>

</xs:complexType>

<xs:complexType name="disqualifiedDirector">

<xs:complexContent>

<xs:extension base="individualRole">

<xs:sequence>

<xs:element name="aliases" type="aliasList" minOccurs="0"/>

<xs:element name="associations" type="associatedCompanyList" minOccurs="0"/>

<xs:element name="disqualificationCriteria" type="disqualificationCriteriaList" minOccurs="0"/>

</xs:sequence>

</xs:extension>

</xs:complexContent>

</xs:complexType>

<xs:complexType name="aliasList">

<xs:sequence>

<xs:element name="aliases" type="xs:string" nillable="true" minOccurs="0" maxOccurs="unbounded"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="associatedCompanyList">

<xs:sequence>

<xs:element name="associations" type="association" nillable="true" minOccurs="0" maxOccurs="unbounded"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="association">

<xs:sequence>

<xs:element name="associatedCompanyName" type="xs:string" minOccurs="0"/>

<xs:element name="associatedCompanyNumber" type="xs:string" minOccurs="0"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="disqualificationCriteriaList">

<xs:sequence>

<xs:element name="criteria" type="disqualificationCriteria" nillable="true" minOccurs="0" maxOccurs="unbounded"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="disqualificationCriteria">

<xs:sequence>

<xs:element name="comments" type="xs:string" minOccurs="0"/>

<xs:element name="criteria" type="xs:string" minOccurs="0"/>

<xs:element name="endDate" type="xs:date" minOccurs="0"/>

<xs:element name="startDate" type="xs:date" minOccurs="0"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="filingCode">

<xs:sequence>

<xs:element name="code" type="xs:string" minOccurs="0"/>

<xs:element name="description" type="xs:string" minOccurs="0"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="attachments">

<xs:sequence>

<xs:element name="attachment" type="attachment" nillable="true" minOccurs="0" maxOccurs="unbounded"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="attachment">

<xs:sequence>

<xs:element name="uri" type="xs:string" minOccurs="0"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="historicAddress">

<xs:complexContent>

<xs:extension base="address">

<xs:sequence>

<xs:element name="effectiveFrom" type="xs:date"/>

<xs:element name="effectiveTo" type="xs:date"/>

</xs:sequence>

</xs:extension>

</xs:complexContent>

</xs:complexType>

<xs:complexType name="entitySearchResults">

<xs:complexContent>

<xs:extension base="baseSearchResults">

<xs:sequence>

<xs:element name="companies" type="enterpriseCompaniesSearchResults" minOccurs="0"/>

<xs:element name="totalResults" type="xs:long" minOccurs="0"/>

</xs:sequence>

</xs:extension>

</xs:complexContent>

</xs:complexType>

<xs:complexType name="baseSearchResults">

<xs:sequence>

<xs:element name="currentPage" type="xs:int"/>

<xs:element name="pageSize" type="xs:int"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="enterpriseCompaniesSearchResults">

<xs:sequence>

<xs:element name="company" type="entitySearchSummary" minOccurs="0" maxOccurs="unbounded"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="entitySearchSummary" abstract="true">

<xs:sequence>

<xs:element name="companyName" type="xs:string"/>

<xs:element name="entityType" type="codedValue" minOccurs="0"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="codedValue">

<xs:sequence>

<xs:element name="code" type="xs:string" minOccurs="0"/>

<xs:element name="description" type="xs:string" minOccurs="0"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="nameReservationSearchResult">

<xs:complexContent>

<xs:extension base="entitySearchSummary">

<xs:sequence>

<xs:element name="nameReservationNumber" type="xs:string" minOccurs="0"/>

</xs:sequence>

</xs:extension>

</xs:complexContent>

</xs:complexType>

<xs:complexType name="entitySearchResult">

<xs:complexContent>

<xs:extension base="entitySearchSummary">

<xs:sequence>

<xs:element name="companyNumber" type="xs:string" minOccurs="0"/>

<xs:element name="companyUri" type="xs:string" minOccurs="0"/>

<xs:element name="incorporationDate" type="xs:anySimpleType" minOccurs="0"/>

<xs:element name="registeredOfficeAddress" type="address" minOccurs="0"/>

<xs:element name="status" type="codedValue" minOccurs="0"/>

</xs:sequence>

</xs:extension>

</xs:complexContent>

</xs:complexType>

<xs:complexType name="roleSearchResults">

<xs:complexContent>

<xs:extension base="baseSearchResults">

<xs:sequence>

<xs:element name="roles" minOccurs="0">

<xs:complexType>

<xs:sequence>

<xs:element name="role" type="role" minOccurs="0" maxOccurs="unbounded"/>

</xs:sequence>

</xs:complexType>

</xs:element>

<xs:element name="totalResults" type="xs:long" minOccurs="0"/>

</xs:sequence>

</xs:extension>

</xs:complexContent>

</xs:complexType>

<xs:complexType name="historicAddresses">

<xs:sequence>

<xs:element name="registeredOfficeAddresses" type="historicAddress" nillable="true" minOccurs="0" maxOccurs="unbounded"/>

<xs:element name="addressForService" type="historicAddress" nillable="true" minOccurs="0" maxOccurs="unbounded"/>

<xs:element name="addressForShareRegister" type="historicAddress" nillable="true" minOccurs="0" maxOccurs="unbounded"/>

<xs:element name="premiseAddress" type="historicAddress" nillable="true" minOccurs="0" maxOccurs="unbounded"/>

<xs:element name="addressForRecords" type="historicAddress" nillable="true" minOccurs="0" maxOccurs="unbounded"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="alias">

<xs:sequence>

<xs:element name="firstName" type="xs:string" minOccurs="0"/>

<xs:element name="lastName" type="xs:string" minOccurs="0"/>

<xs:element name="middleName" type="xs:string" minOccurs="0"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="filingRequirement">

<xs:sequence>

<xs:element name="archived" type="xs:boolean"/>

<xs:element name="attachments" type="attachments" minOccurs="0"/>

<xs:element name="filingCode" type="filingCode" minOccurs="0"/>

<xs:element name="registrationDate" type="xs:date" minOccurs="0"/>

<xs:element name="url" type="xs:string" minOccurs="0"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="enterpriseCompaniesSearchResultList">

<xs:complexContent>

<xs:extension base="arrayList">

<xs:sequence/>

</xs:extension>

</xs:complexContent>

</xs:complexType>

<xs:complexType name="arrayList">

<xs:complexContent>

<xs:extension base="abstractList">

<xs:sequence/>

</xs:extension>

</xs:complexContent>

</xs:complexType>

<xs:complexType name="abstractList" abstract="true">

<xs:complexContent>

<xs:extension base="abstractCollection">

<xs:sequence/>

</xs:extension>

</xs:complexContent>

</xs:complexType>

<xs:complexType name="abstractCollection" abstract="true">

<xs:sequence/>

</xs:complexType>

<xs:complexType name="additionalCompanyInformation">

<xs:sequence>

<xs:element name="bicCode" type="codedValue" minOccurs="0"/>

<xs:element name="nzsxCode" type="xs:string" minOccurs="0"/>

<xs:element name="premiseAddress" type="address" minOccurs="0"/>

<xs:element name="publicEmailAddress" type="xs:string" minOccurs="0"/>

<xs:element name="publicPhoneNumber" type="xs:string" minOccurs="0"/>

<xs:element name="tradingName" type="xs:string" minOccurs="0"/>

<xs:element name="websiteAddress" type="xs:string" minOccurs="0"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="directorList">

<xs:sequence>

<xs:element name="director" type="individualRole" maxOccurs="unbounded"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="addressList">

<xs:sequence>

<xs:element name="address" type="address" minOccurs="0" maxOccurs="unbounded"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="shareParcelList">

<xs:sequence>

<xs:element name="shareParcel" type="shareParcel" minOccurs="0" maxOccurs="unbounded"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="shareParcel">

<xs:sequence>

<xs:element name="shareholders" type="role" minOccurs="0" maxOccurs="unbounded"/>

<xs:element name="numberOfShares" type="xs:long"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="documentList">

<xs:sequence>

<xs:element name="filingRequirement" type="filingRequirement" maxOccurs="unbounded"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="shareholderList">

<xs:sequence>

<xs:element name="shareholder" type="role" maxOccurs="unbounded"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="entityDetail">

<xs:sequence>

<xs:element name="companyName" type="xs:string"/>

<xs:element name="companyNumber" type="xs:string"/>

<xs:element name="companyUri" type="xs:string"/>

<xs:element name="singleBusinessNumber" type="xs:string" minOccurs="0"/>

<xs:element name="incorporationDate" type="xs:date"/>

<xs:element name="entityType" type="codedValue"/>

<xs:element name="status" type="codedValue"/>

<xs:element name="annualReturnFilingMonth" type="xs:long"/>

<xs:element name="financialReportingFilingMonth" type="xs:long" minOccurs="0"/>

<xs:element name="externalAdministration" type="externalAdministration" nillable="true" minOccurs="0" maxOccurs="unbounded"/>

<xs:element name="australianCompanyNumber" type="xs:string" minOccurs="0"/>

<xs:element name="countryOfOrigin" type="codedValue" minOccurs="0"/>

<xs:element name="inRemoval" type="xs:boolean"/>

<xs:element name="hasConstitutionFiled" type="xs:boolean"/>

<xs:element name="reRegistered" type="xs:boolean"/>

<xs:element name="reRegistrationDate" type="xs:date" minOccurs="0"/>

<xs:element name="totalNumberOfShares" type="xs:long"/>

<xs:element name="extensiveShareholding" type="xs:boolean"/>

<xs:element name="registeredOfficeAddress" type="address"/>

<xs:element name="addressForService" type="address" minOccurs="0"/>

<xs:element name="previousCompanyNames" type="historicName" nillable="true" minOccurs="0" maxOccurs="unbounded"/>

<xs:element name="previousCompanyStatuses" type="historicCodedValue" nillable="true" minOccurs="0" maxOccurs="unbounded"/>

<xs:element name="addressForRecords" type="addressList" minOccurs="0"/>

<xs:element name="addressForShareRegister" type="addressList" minOccurs="0"/>

<xs:element name="personsAuthorisedToAcceptService" minOccurs="0">

<xs:complexType>

<xs:sequence>

<xs:element name="personsAuthorisedToAcceptService" type="individualRole" minOccurs="0" maxOccurs="unbounded"/>

</xs:sequence>

</xs:complexType>

</xs:element>

<xs:element name="historicAddresses" type="historicAddresses" minOccurs="0"/>

<xs:element name="activeDirectors" type="directorList"/>

<xs:element name="inactiveDirectors" minOccurs="0">

<xs:complexType>

<xs:sequence>

<xs:element name="director" type="individualRole" minOccurs="0" maxOccurs="unbounded"/>

</xs:sequence>

</xs:complexType>

</xs:element>

<xs:element name="shareParcels" type="shareParcelList" minOccurs="0"/>

<xs:element name="shareholder" type="role" minOccurs="0" maxOccurs="unbounded"/>

<xs:element name="documents" type="documentList"/>

<xs:element name="additionalCompanyInformation" type="additionalCompanyInformation" minOccurs="0"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="historicName">

<xs:sequence>

<xs:element name="companyName" type="xs:string" minOccurs="0"/>

<xs:element name="dateChanged" type="xs:date"/>

</xs:sequence>

</xs:complexType>

<xs:complexType name="historicCodedValue">

<xs:complexContent>

<xs:extension base="codedValue">

<xs:sequence>

<xs:element name="dateChanged" type="xs:date"/>

</xs:sequence>

</xs:extension>

</xs:complexContent>

</xs:complexType>

</xs:schema>

# Appendix 1

## List of Permitted Characters for Searching

|  |  |  |
| --- | --- | --- |
| Value | Description | Hex conversion |
| A – Z |  |  |
| 0 – 9 |  |  |
|  | Diacritics and Maori macrons (please refer to table below) |  |
|  | Blank | %20 |
| ! | Exclamation mark | %21 |
| “ | Double quote | %22 |
| # | Hash | %23 |
| $ | Dollar sign | %24 |
| % | Percentage sign | %25 |
| & | Ampersand | %26 |
| ‘ | Single quote | %27 |
| ( | Open bracket | %28 |
| ) | Close bracket | %29 |
| \* | Asterisk | %2A |
| + | Plus sign | %2B |
| , | Comma | %2C |
| - | Dash (minus sign) |  |
| . | Dot (period) |  |
| / | Forward slash | %2F |
| : | Colon | %3A |
| ; | Semi-colon | %3B |
| < | Less than sign | %3C |
| > | Greater than sign | %3E |
| = | Equal sign | %3D |
| ? | Question mark | %3F |
| @ | At sign | %40 |
| [ | Open square bracket | %5B |
| ] | Close square bracket | %5D |
| \ | Back slash | %5C |
| ^ | Caret | %5E |
| \_ | Underscore |  |
| ` | Grave accent | %60 |
| { | Open curly bracket | %7B |
| } | Close curly bracket | %7D |
| | | Pipe | %7C |
| ~ | tilde |  |

## List of Diacritics

Please note that only Maori macrons/diacritics are accepted. They follow the hex code below:

|  |  |  |
| --- | --- | --- |
| Value | Description | Hex conversion |
| ā | a latin small letter a with macron | u0101 |
| À | A latin capital letter a with breve | u0102 |
| Ē | E latin capital letter e with macron | u0112 |
| ē | e latin small letter e with macron | u0113 |
| Ī | I latin capital letter i with macron | u012A |
| ī | i latin small letter i with macron | u012B |
| Ō | O latin capital letter o with macron | u014C |
| ō | o latin small letter o with macron | u014D |
| Ū | U latin capital letter u with macron | u016A |
| ū | u latin small letter u with macron | u016B |

## List of all Entity Status Values

|  |  |
| --- | --- |
| Parameter Value | Description |
| 2 | Refused |
| 3 | Already On the Register |
| 4 | Similar Name On Register |
| 5 | Similar To Approved Name |
| 6 | Name Already Approved |
| 7 | Refusal Other |
| 8 | Prohibited Name |
| 20 | Withdrawn Approval |
| 30 | Consent Required |
| 33 | Other Consent |
| 34 | Pending Restoration |
| 35 | Awaiting Registration |
| 36 | Awaiting Registration Hold |
| 37 | Pre Pending |
| 38 | Deleted - Incorporation |
| 40 | Reserved |
| 41 | Under Inc With Client |
| 42 | Inc Submitted By Client |
| 43 | Pending Consent Correctn |
| 44 | Under Inc With B&Rb |
| 45 | Awaiting Consent Hold |
| 46 | Pending Registration |
| 47 | Pending - Incorporation |
| 48 | Pending Hold |
| 49 | Expired - Incorporation |
| 50 | Registered |
| 55 | Voluntary Administration |
| 56 | Voluntary Administration & In Rec |
| 57 | In Receivership & In Voluntary Administration |
| 60 | In Liq |
| 61 | In Liq and Rec |
| 62 | Struck Off & In Liq. |
| 63 | In Receivership & In Liquidation & In Voluntary Administration |
| 64 | In Liquidation & In Receivership & In Voluntary Administration |
| 65 | In Liquidation & Voluntary Administration |
| 66 | In Liquidation & Voluntary Administration & In Receivership |
| 70 | In Rec |
| 71 | In Rec. & In Liq. |
| 72 | In Statutory Mgmt. |
| 73 | In Admin. |
| 80 | Struck Off |

## List of All Entity Body Type Values

|  |  |
| --- | --- |
| Parameter Value | Description |
| LTD | Limited Company |
| UNTLD | UnLimited Company |
| ASIC | Australian Company |
| NON\_ASIC | Overseas (non Australian) Company |
| COOP | Co-operative Company |

## List of All Country Code Values

Country codes returned as part of the addresses adhere to the [ISO\_3166-1 address standard](http://en.wikipedia.org/wiki/ISO_3166-1).

|  |  |
| --- | --- |
| Value | Country Description |
| AD | ANDORRA |
| AE | UNITED ARAB EMIRATES |
| AF | AFGHANISTAN |
| AG | ANTIGUA AND BARBUDA |
| AI | ANGUILLA |
| AL | ALBANIA |
| AM | ARMENIA |
| AN | NETHERLANDS ANTILLES |
| AO | ANGOLA |
| AQ | ANTARCTICA |
| AR | ARGENTINA |
| AS | AMERICAN SAMOA |
| AT | AUSTRIA |
| AU | AUSTRALIA |
| AW | ARUBA |
| AX | ALAND ISLANDS |
| AZ | AZERBAIJAN |
| BA | BOSNIA AND HERZEGOVINA |
| BB | BARBADOS |
| BD | BANGLADESH |
| BE | BELGIUM |
| BF | BURKINA FASO |
| BG | BULGARIA |
| BH | BAHRAIN |
| BI | BURUNDI |
| BJ | BENIN |
| BL | SAINT BARTHELEMY |
| BM | BERMUDA |
| BN | BRUNEI DARUSSALAM |
| BO | BOLIVIA |
| BR | BRAZIL |
| BS | BAHAMAS |
| BT | BHUTAN |
| BV | BOUVET ISLAND |
| BW | BOTSWANA |
| BY | BELARUS |
| BZ | BELIZE |
| CA | CANADA |
| CC | COCOS (KEELING) ISLANDS |
| CD | CONGO, THE DEMOCRATIC REPUBLIC OF THE |
| CF | CENTRAL AFRICAN REPUBLIC |
| CG | CONGO |
| CH | SWITZERLAND |
| CI | COTE D'IVOIRE |
| CK | COOK ISLANDS |
| CL | CHILE |
| CM | CAMEROON |
| CN | CHINA |
| CO | COLOMBIA |
| CR | COSTA RICA |
| CU | CUBA |
| CV | CAPE VERDE |
| CX | CHRISTMAS ISLAND |
| CY | CYPRUS |
| CZ | CZECH REPUBLIC |
| DE | GERMANY |
| DJ | DJIBOUTI |
| DK | DENMARK |
| DM | DOMINICA |
| DO | DOMINICAN REPUBLIC |
| DZ | ALGERIA |
| EC | ECUADOR |
| EE | ESTONIA |
| EG | EGYPT |
| EH | WESTERN SAHARA |
| ER | ERITREA |
| ES | SPAIN |
| ET | ETHIOPIA |
| FI | FINLAND |
| FJ | FIJI |
| FK | FALKLAND ISLANDS (MALVINAS) |
| FM | MICRONESIA, FEDERATED STATES OF |
| FO | FAROE ISLANDS |
| FR | FRANCE |
| GA | GABON |
| GB | UNITED KINGDOM |
| GD | GRENADA |
| GE | GEORGIA |
| GF | FRENCH GUIANA |
| GG | GUERNSEY |
| GH | GHANA |
| GI | GIBRALTAR |
| GL | GREENLAND |
| GM | GAMBIA |
| GN | GUINEA |
| GP | GUADELOUPE |
| GQ | EQUATORIAL GUINEA |
| GR | GREECE |
| GS | SOUTH GEORGIA AND THE SOUTH SANDWICH ISLANDS |
| GT | GUATEMALA |
| GU | GUAM |
| GW | GUINEA-BISSAU |
| GY | GUYANA |
| HK | HONG KONG |
| HM | HEARD ISLAND AND MCDONALD ISLANDS |
| HN | HONDURAS |
| HR | CROATIA |
| HT | HAITI |
| HU | HUNGARY |
| ID | INDONESIA |
| IE | IRELAND |
| IL | ISRAEL |
| IM | ISLE OF MAN |
| IN | INDIA |
| IO | BRITISH INDIAN OCEAN TERRITORY |
| IQ | IRAQ |
| IR | IRAN, ISLAMIC REPUBLIC OF |
| IS | ICELAND |
| IT | ITALY |
| JE | JERSEY |
| JM | JAMAICA |
| JO | JORDAN |
| JP | JAPAN |
| KE | KENYA |
| KG | KYRGYZSTAN |
| KH | CAMBODIA |
| KI | KIRIBATI |
| KM | COMOROS |
| KN | SAINT KITTS AND NEVIS |
| KP | KOREA, DEMOCRATIC PEOPLE'S REPUBLIC OF |
| KR | KOREA, REPUBLIC OF |
| KW | KUWAIT |
| KY | CAYMAN ISLANDS |
| KZ | KAZAKHSTAN |
| LA | LAO PEOPLE'S DEMOCRATIC REPUBLIC |
| LB | LEBANON |
| LC | SAINT LUCIA |
| LI | LIECHTENSTEIN |
| LK | SRI LANKA |
| LR | LIBERIA |
| LS | LESOTHO |
| LT | LITHUANIA |
| LU | LUXEMBOURG |
| LV | LATVIA |
| LY | LIBYAN ARAB JAMAHIRIYA |
| MA | MOROCCO |
| MC | MONACO |
| MD | MOLDOVA, REPUBLIC OF |
| ME | MONTENEGRO |
| MF | SAINT MARTIN |
| MG | MADAGASCAR |
| MH | MARSHALL ISLANDS |
| MK | MACEDONIA, THE FORMER YUGOSLAV REPUBLIC OF |
| ML | MALI |
| MM | MYANMAR |
| MN | MONGOLIA |
| MO | MACAO |
| MP | NORTHERN MARIANA ISLANDS |
| MQ | MARTINIQUE |
| MR | MAURITANIA |
| MS | MONTSERRAT |
| MT | MALTA |
| MU | MAURITIUS |
| MV | MALDIVES |
| MW | MALAWI |
| MX | MEXICO |
| MY | MALAYSIA |
| MZ | MOZAMBIQUE |
| NA | NAMIBIA |
| NC | NEW CALEDONIA |
| NE | NIGER |
| NF | NORFOLK ISLAND |
| NG | NIGERIA |
| NI | NICARAGUA |
| NL | NETHERLANDS |
| NO | NORWAY |
| NP | NEPAL |
| NR | NAURU |
| NU | NIUE |
| NZ | NEW ZEALAND |
| OM | OMAN |
| PA | PANAMA |
| PE | PERU |
| PF | FRENCH POLYNESIA |
| PG | PAPUA NEW GUINEA |
| PH | PHILIPPINES |
| PK | PAKISTAN |
| PL | POLAND |
| PM | SAINT PIERRE AND MIQUELON |
| PN | PITCAIRN |
| PR | PUERTO RICO |
| PS | PALESTINIAN TERRITORY, OCCUPIED |
| PT | PORTUGAL |
| PW | PALAU |
| PY | PARAGUAY |
| QA | QATAR |
| RE | REUNION |
| RO | ROMANIA |
| RS | SERBIA |
| RU | RUSSIAN FEDERATION |
| RW | RWANDA |
| SA | SAUDI ARABIA |
| SB | SOLOMON ISLANDS |
| SC | SEYCHELLES |
| SD | SUDAN |
| SE | SWEDEN |
| SG | SINGAPORE |
| SH | SAINT HELENA |
| SI | SLOVENIA |
| SJ | SVALBARD AND JAN MAYEN |
| SK | SLOVAKIA |
| SL | SIERRA LEONE |
| SM | SAN MARINO |
| SN | SENEGAL |
| SO | SOMALIA |
| SR | SURINAME |
| ST | SAO TOME AND PRINCIPE |
| SV | EL SALVADOR |
| SY | SYRIAN ARAB REPUBLIC |
| SZ | SWAZILAND |
| TC | TURKS AND CAICOS ISLANDS |
| TD | CHAD |
| TF | FRENCH SOUTHERN TERRITORIES |
| TG | TOGO |
| TH | THAILAND |
| TJ | TAJIKISTAN |
| TK | TOKELAU |
| TL | TIMOR-LESTE |
| TM | TURKMENISTAN |
| TN | TUNISIA |
| TO | TONGA |
| TR | TURKEY |
| TT | TRINIDAD AND TOBAGO |
| TV | TUVALU |
| TW | TAIWAN, PROVINCE OF CHINA |
| TZ | TANZANIA, UNITED REPUBLIC OF |
| UA | UKRAINE |
| UG | UGANDA |
| UM | UNITED STATES MINOR OUTLYING ISLANDS |
| US | UNITED STATES |
| UY | URUGUAY |
| UZ | UZBEKISTAN |
| VA | HOLY SEE (VATICAN CITY STATE) |
| VC | SAINT VINCENT AND THE GRENADINES |
| VE | VENEZUELA, BOLIVARIAN REPUBLIC OF |
| VG | VIRGIN ISLANDS, BRITISH |
| VI | VIRGIN ISLANDS, U.S. |
| VN | VIET NAM |
| VU | VANUATU |
| WF | WALLIS AND FUTUNA |
| WS | SAMOA |
| YE | YEMEN |
| YT | MAYOTTE |
| ZA | SOUTH AFRICA |
| ZM | ZAMBIA |
| ZW | ZIMBABWE |

# Appendix 2 Code Samples

## .NET

            string accessKey = "T4WZUVoufkaGfsrK";

            string secretAuthenticationKey = "86MVgkk3fdiLlzPCXaXDGCn22DSqEVFJ8GNzKGED";

            Uri baseUri = new Uri("<http://www.dev.business.govt.nz>", UriKind.Absolute);

            Uri requestUri = new Uri(baseUri, "/data/app/ws/rest/companies/entity/v1.0/1277262");

            DateTime requestDate = new DateTime(2012, 2, 22, 2, 23, 0, DateTimeKind.Utc);  // to match the example. Normally use DateTime.UtcNow

            HttpWebRequest request = (HttpWebRequest)WebRequest.Create(requestUri);

            request.Date = requestDate;

            request.Accept = "application/xml,text/html,application/xhtml+xml,application/xml;q=0.9,\*/\*;q=0.8";  // to match the example

            string stringToSign = request.Method + "\n" + request.Host + "\n" + request.RequestUri.AbsolutePath + "\n"

                                    + request.Headers[HttpRequestHeader.Date] + "\n" + accessKey + "\n"

                                    + request.Headers[HttpRequestHeader.Accept] + "\n";

            byte[] hashBytes = null;

            byte[] secretKeyBytes = Encoding.UTF8.GetBytes(secretAuthenticationKey);

            byte[] stringToSignBytes = Encoding.UTF8.GetBytes(stringToSign);

            using (HMAC hmac = new HMACSHA256(secretKeyBytes))

            {

                hashBytes = hmac.ComputeHash(stringToSignBytes);

            }

            string hash = Convert.ToBase64String(hashBytes);

            // at this point hash contains "QSYqcxEF6pT0Ds5cZRI6Tz708CjLomF0dZ0LDzX1IFw="

            request.Headers["Authorization"] = accessKey + ":" + hash;

## PHP

<?php

class CompanySearchApi

{

private $url='http://www.EAT.business.govt.nz/data/app/ws/rest/companies/entity/search/v1.0';

private $auth\_key="HjYwDDg334jNr3ho";

private $secret\_auth\_key="lYGuVrWeOWVLmojZGJ5NI7phBrMs4A0LFJDNUK9x";

private $base\_domain="www.EAT.business.govt.nz";

private $request\_auth\_key = "";

public function \_\_construct(){

}

public function search\_company($search\_string){

//$search\_string = "optimizerhq";

$request = '';

$request .= "GET\n";

$request .= "www.EAT.business.govt.nz\n";

$request .= "/data/app/ws/rest/companies/entity/search/v1.0/".$search\_string."\n";

$request .= date("D, d M Y H:i:s")." NZT\n";

$request .= $this->auth\_key."\n";

$request .= "application/json\n";

$this->request\_auth\_key = base64\_encode(hash('sha256',$request,TRUE));

$temp\_url = $this->url."/".$search\_string;

$ch = curl\_init();

$headers = array("Authorization:".$this->request\_auth\_key);

curl\_setopt($ch, CURLOPT\_HTTPHEADER, $headers);

curl\_setopt($ch,CURLOPT\_URL,$temp\_url);

//var\_dump($temp\_url);

curl\_setopt($ch,CURLOPT\_HEADER,0);

curl\_setopt($ch,CURLOPT\_RETURNTRANSFER,1);

curl\_setopt($ch,CURLOPT\_SSL\_VERIFYPEER,0);

curl\_setopt($ch,CURLOPT\_SSL\_VERIFYHOST,0);

curl\_setopt($ch, CURLOPT\_PROXYPORT, '8080');

$html = curl\_exec($ch);

curl\_close($ch);

var\_dump($html);

return $html;

}

}

?>

## Groovy Script

import java.text.SimpleDateFormat

import javax.crypto.Mac

import javax.crypto.spec.SecretKeySpec

import java.util.regex.Matcher

import java.util.regex.Pattern

import org.apache.commons.codec.binary.Base64

import com.eviware.soapui.support.types.StringToStringMap

import com.eviware.soapui.\*

//log.info "context is ${context}"

//log.info "request is ${request}"

log.info "URI is ${context.httpMethod.getURI()}"

addHeaders(request, context)

//------------

// Functions

//------------

def addHeaders(restTestRequest, context){

accept = "application/xml"

httpDate = new SimpleDateFormat('EEE, dd MMM yyyy HH:mm:ss zzz').format(new Date())

if(restTestRequest.endpoint=~'eat'){

secretKey = restTestRequest.operation.interface.project.getPropertyValue("EATsecretKey")

accessKey = restTestRequest.operation.interface.project.getPropertyValue("EATaccessKey")

//log.info "TEST Resquest"

}else{

secretKey = restTestRequest.operation.interface.project.getPropertyValue("secretKey")

accessKey = restTestRequest.operation.interface.project.getPropertyValue("accessKey")

//log.info "LIVE Resquest"

}

pathToSign = context.httpMethod.getURI()

//log.info "pathToSign is ${pathToSign}"

source = createSource(restTestRequest,pathToSign, accessKey, accept, httpDate)

//log.info "Source is ${source}"

signature = createSignature(secretKey, source)

headers = new StringToStringMap()

headers.put("Authorization", accessKey + ":" + signature)

headers.put("Date", httpDate)

headers.put("Accept", accept)

//restTestRequest.setPath("/data/app/ws/rest/companies/role/search/v1.0")

restTestRequest.setRequestHeaders(headers)

//log.info "after setting headers"

}

def createSource(restTestRequest, path, accessKey, accept, httpDate){

//log.info "before endpoint"

endpoint = restTestRequest.endpoint

//log.info "endpoint is ${endpoint}"

//log.info "before uri, path is ${path}"

uri = path.toString().replaceAll(endpoint,"")

//log.info "${uri}"

return "${restTestRequest.method}\n" +

"${endpoint.replaceAll("http://", "")}\n" +

"$uri\n" +

"$httpDate\n" +

"$accessKey\n" +

"$accept\n"

}

def createSignature(secretKey, source){

Mac hmac = Mac.getInstance('HmacSHA256')

hmac.init(new SecretKeySpec(secretKey.getBytes(), 'HmacSHA256'))

rawHmac = hmac.doFinal(source.getBytes());

return new String(Base64.encodeBase64(rawHmac))

}