

Inland Revenue

Build Pack: Software Intermediation Service

Date: 24/08/2020

Contents

1 Overview	3
1.1 This solution	3
1.2 Intended audience	3
1.3 Related services	3
1.3.1 Transaction Data Services	3
1.3.2 Identity and Access Services	4
1.4 Prerequisites	4
1.4.1 Mutual Transport Layer Security and certificates	4
2 Solution design	5
2.1 Architecture	5
2.2 Service scope	5
2.3 Messaging	5
2.4 Security	7
3 Operations	10
3.1 Link	12
3.2 Delink	14
3.3 RetrieveClientList	16
4 End points, schemas and WSDLs	18
4.1 End points	18
4.2 Schemas	18
4.3 WSDLs	18
5 Response codes	19
5.1 Generic gateway response codes	19
5.2 Generic software intermediation response codes	20
6 Glossary	22
7 Change log	25

1 Overview

1.1 This solution

Inland Revenue has a suite of digital services available for consumption by software providers that supports efficient, electronic business interactions with Inland Revenue. The Software Intermediation Service described in this build pack document forms part of a suite of Gateway Services.

The Software Intermediation Service provides the ability for software providers to link to their Tax Agency clients or directly to their Customers in order to be provided an initial full data load and daily bulk file updates of their transaction data. The Software Intermediation Service allows these links to be established, queried, or removed. There is a separate Intermediation Service that addresses the links between business intermediaries like tax agents and their client lists and their customer accounts. The actual dataset resulting in bulk files from a software intermediation link will be impacted by those intermediation links but the links themselves are maintained separately. This is explained in more detail in the Transaction Data Services (TDS) Overview build pack.

This document is intended to provide the technical details required to support the consumption of this Gateway Service. It describes the architecture of the technical solution, schemas, end points and also its interaction with other build packs that cover different aspects of Gateway Services.

Before continuing, please consult
www.ird.govt.nz/digital-service-providers/services-catalogue
for business-level context, use cases and links to relevant policy.
The information available here explains how to integrate with
Inland Revenue's services.

1.2 Intended audience

The solution outlined in this document is intended to be used by technical teams and development staff. It describes the technical interactions, including responses, provided by the Software Intermediation Service.

The reader is assumed to have a suitable level of technical knowledge in order to comprehend the information provided. A range of technical terms and abbreviations are used throughout this document, and while most of these will be understood by the intended readers, a [glossary](#) is provided at the end.

1.3 Related services

The following application programming interfaces (APIs) complement this Gateway Service. Instructions on where to find the build packs for these APIs can be found in [section 4](#) of this document.

1.3.1 Transaction Data Services

Transaction Data Services are intended to support software providers in their transition from Tax Agent Web Services to the use of TDS. The Transaction Data Services Overview and

Transition build pack provides an overview of TDS, describes the data which will be made available through the services and the processes, as well as giving use cases for how these services will be employed.

1.3.2 Identity and Access Services

Identity and Access Services (IAS) are used to authenticate access. Authentication tokens will need to be retrieved via IAS prior to making calls to the Software Intermediation Service.

This build pack was written using information from version 1.5 of the IAS build pack.

1.4 Prerequisites

Party	Requirement	Description
Software Provider	Acquire a X.509 certificate from a certificate authority for the Test and Production environments	This is required when using mutual TLS with cloud-based software providers. Note that the same certificate cannot be used for the Test and Production environments.

1.4.1 Mutual Transport Layer Security and certificates

Mutual transport layer security (TLS) is implemented for the Software Intermediation Service. This requires the use of a publicly-issued X509 certificate from one of the trusted certificate authorities. Inland Revenue does not issue certificates to external vendors for web service security implementations.

Inland Revenue has the following minimum requirements for accepting public X509 keys:

1. Minimum Key Length: 2048
2. Signature Algorithm: SHA256[RSA]
3. Self-signed certificates are not accepted
4. Certificates issued by a private/internal certificate authority are not accepted.

In general, shorter-lived certificates offer a better security posture since the impact of key compromise is less severe but there is no minimum requirement for certificate expiry periods.

Below is a list for examples of certificate authority providers with no recommendations or rankings incorporated. It is recommended that a business researches which certificate authority meets their requirements.

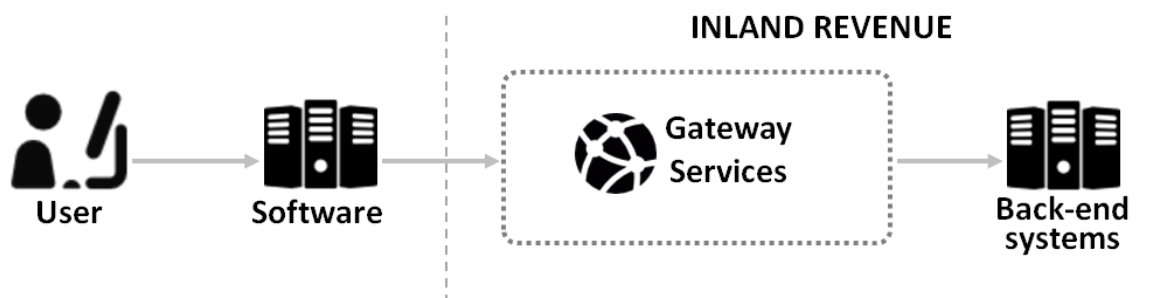
- [Comodo](#)
- [GeoTrust](#)
- [DigiCert](#)
- [GlobalSign](#)
- [Symantec](#)
- [Thawte](#)
- [IdenTrust](#)
- [Entrust](#)
- [Network Solutions](#)
- [RapidSSL](#)
- [Entrust Datacard](#)
- [GoDaddy](#).

2 Solution design

2.1 Architecture

Inland Revenue is offering a suite of web services in order to facilitate interactions with Inland Revenue via software packages. The Gateway Services suite will be used by approved software providers to facilitate everything from registration activities, filing returns, making payments and other service offerings in order to allow customers to interact with Inland Revenue.

The diagram below illustrates the flow of data from the customer to Inland Revenue.



The online WSDLs for the Gateway Services define an 'any' XML request and response structure, which then relies on a group of XSDs to define the data structure of those requests and responses. Each request and response type will define a lower, 'wrapper' element. To simplify analysis and code generation, a development oriented version of the WSDL and XSDs is provided with the build pack that has the any elements replaced with relevant types.

Any malformed XML will instantly be rejected by the Gateway Services prior to any schema validation.

2.2 Service scope

The Software Intermediation Service supports the following operations:

- **Link:** This service is used to create a link between a software intermediary and a client.
- **Delink:** This service is used to cease a link between the above parties.
- **RetrieveClientList:** This service is used to retrieve a list of the software intermediary's clients.

2.3 Messaging

All SOAP messages require a SOAP header containing the **Action:** parameter, as well as a SOAP body containing a structured XML payload. Please refer to the WSDL for the correct addresses.

The online WSDLs for the Gateway Services define an 'any' XML request and response structure, which then relies on a group of XSDs to define the data structure of those requests and responses. Each request and response type will define a lower, 'wrapper' element. To simplify analysis and code generation, a development-oriented version of the WSDL and XSDs is provided with the build pack that has the 'any' elements replaced with relevant types.

The Gateway Services allow the consumption of any structured XML payload but will be validated against the Inland Revenue-published XSDs.

This is a late binding validation, performed after authentication has been reviewed. The message structure of these services is a simple request/response. The XML request will be checked for well-formed XML before the schema validation. Responses to these requests will be in XML format as well and will be defined in the same schemas that define the requests.

Any XML submissions in the SOAP body that do not meet the provided schemas will not be accepted by the Gateway Services. Incorrect namespaces will also fail validation against the published schemas.

Example SOAP request structure

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
  xmlns:sft="https://services.ird.govt.nz/GWS/SoftwareIntermediation/"
  xmlns:gcl="https://services.ird.govt.nz/GWS/SoftwareIntermediation/types/RetrieveClientListRequest"
  xmlns:a="http://www.w3.org/2005/08/addressing">
  <soap:Header>
    <a:Action>https://services.ird.govt.nz/GWS/SoftwareIntermediation/SoftwareIntermediation/Operation</a:Action>
  </soap:Header>
  <soap:Body>
    <sft:RetrieveClientList>
      <sft:RetrieveClientListRequestMsg>
        <rcl:RetrieveClientListRequestWrapper>
          <RetrieveClientListRequest xmlns:xsi...
            <...XML payload...>
          </RetrieveClientListRequest>
        </rcl:RetrieveClientListRequestWrapper>
      </sft:RetrieveClientListRequestMsg>
    </sft:RetrieveClientList>
  </soap:Body>
</soap:Envelope>
```

Example SOAP response structure

```
<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
  xmlns:a=http://www.w3.org/2005/08/addressing
  xmlns:si=https://services.ird.govt.nz/GWS/SoftwareIntermediation/
  xmlns:b="https://services.ird.govt.nz/GWS/SoftwareIntermediation/types/RetrieveClientListResponse"
  xmlns:i=http://www.w3.org/2001/XMLSchema-instance
  xmlns:cmn="urn:www.ird.govt.nz/GWS/types/Common.v1">
  <s:Header>
    <a:Action s:mustUnderstand="1">
      https://services.ird.govt.nz/GWS/SoftwareIntermediation/SoftwareIntermediation/RetrieveClientListResponse
    </a:Action>
  </s:Header>
  <s:Body>
    <si:RetrieveClientListResponse >
      <si:RetrieveClientListResult>
        <b:RetrieveClientListResponseWrapper>
          <cmn:RetrieveClientListResponse>
            <cmn:statusMessage>
              <cmn:statusCode>0</statusCode>
              <cmn:errorMessage/>
            </cmn:statusMessage>
          </cmn:RetrieveClientListResponse>
        </b:RetrieveClientListResponseWrapper>
      </si:RetrieveClientListResult>
    </si:RetrieveClientListResponse>
  </s:Body>
</s:Envelope>
```

```

    </b:RetrieveClientListResponseWrapper>
  </si:RetrieveClientListResult>
</si:RetrieveClientListResponse>
</s:Body>
</s:Envelope>

```

2.4 Security

Gateway Services requests are access-controlled using an OAuth token that identifies the user making the request. Users will authenticate using their Inland Revenue myIR credentials. For instructions on how to acquire an OAuth token, review the Identity and access build pack. For TDS Real Time web service requests, an OAuth access token is required in the HTTP header. Authorisation for using the Gateway Services is defined in the permissions set in myIR.

Permissions will reflect those granted in myIR. For example, if a user does not have permission to file a return online, they will not be able to file a return via Gateway Services either. This applies to users who are granted access as staff inside an organisation or as staff in a tax agency.

The Gateway Services use an HTTPS transport layer, with HTTP1.1 transport protocol supported.

The Gateway Services also use the SOAP version 1.2 protocol.

The SOAP service contract is published using WSDL version 1.1.

Regarding transport layer security (TLS), note that while TLS1.3 is now an industry standard, it is not yet widely adopted, as doing so requires upgrades to perimeter security devices and software. Inland Revenue will upgrade to TLS1.3 once it is adopted widely enough, and where practical, external software partners should also anticipate upgrading to this version. TLS1.0 and TLS1.1 are not supported by myIR or Gateway Services.

Inland Revenue requires the following ciphers and key strengths to be used:

Encryption:	Advanced Encryption Standard (AES)	FIPS 197	256-bit key
Hashing:	Secure Hash Algorithm (SHA-2)	FIPS 180-3	SHA-256

There will be two end points, which are summarised in the bullet points below (the table immediately afterwards provides more detail):

1. There is an end point to which software providers' centralised **cloud** locations can connect. This will involve mutual TLS certificates that need to be exchanged during the onboarding phase. On the cloud end point, Inland Revenue has controls to shield software providers from issues caused by heavy usage from other providers.
2. For software providers connecting from **desktops**, there is a separate end point that does not use mutual TLS. For this service, certificates do not need to be exchanged during onboarding. On the desktop end point Inland Revenue has less ability to shield consumers of the service from heavy usage by other Inland Revenue partners.

	End point for cloud-based connections	End point for desktop connections
Purpose	<ul style="list-style-type: none"> This is the default end point to connect software providers to the Gateway Services 	<ul style="list-style-type: none"> Additional end point provided to facilitate connecting from desktops which might be high volumes of sources addresses, transient DHCP addresses, not realistically associated with client side TLS certificates, not individually onboarded to setup certificate trust
Client application type	<ul style="list-style-type: none"> Cloud applications 	<ul style="list-style-type: none"> Desktop/native applications For connecting from multiple decentralised clients
Constraints	<ul style="list-style-type: none"> Only for source locations with client-side TLS certificates On the cloud end point Inland Revenue has controls to shield software providers from issues caused by heavy usage from other providers 	<ul style="list-style-type: none"> Less scalable Subject to tighter security controls On the desktop end point Inland Revenue has less ability to shield consumers of the service from heavy usage by others OAuth2 refresh tokens will not be offered to desktop clients
Mutual TLS	<ul style="list-style-type: none"> Inland Revenue explicitly trusts the certificate the software provider associates with the TLS connection as client for Mutual TLS connections and uses it to identify the software provider in conjunction with the web service identification below 	<ul style="list-style-type: none"> Server-side certificates only
Minimum TLS version	<ul style="list-style-type: none"> 1.2 	<ul style="list-style-type: none"> 1.0(+)
URL	<ul style="list-style-type: none"> Contains .../gateway/.. 	<ul style="list-style-type: none"> Contains .../gateway2/..
Port	<ul style="list-style-type: none"> 4046 	<ul style="list-style-type: none"> 443 (Default https port)
Web service consumer identification	<ul style="list-style-type: none"> Each software provider is given a software platform ID during onboarding. This ID is of type Customer ID and is independent of the end point. To be identified in web service calls related to getting an OAuth token each cloud application will be given identity and access system 	<ul style="list-style-type: none"> Each software provider is given a software platform ID during onboarding. This ID is of type Customer ID and is not specific to the end point. Desktop clients will be given different identity and access system client_id/client_secret credentials to cloud application clients

	End point for cloud-based connections	End point for desktop connections
	client_id/client_secret credentials during onboarding to allow it to get OAuth tokens to call this end point <ul style="list-style-type: none"> The mutual TLS certificate is used to identify the service provider 	
Firewalling in production	<ul style="list-style-type: none"> No IP address restrictions Access limited by certificate enrolment 	<ul style="list-style-type: none"> No IP address restrictions
Firewalling in non-production environments	<ul style="list-style-type: none"> No IP address restrictions Access limited by certificate enrolment 	<ul style="list-style-type: none"> Firewalled—IP whitelisting needed

Delegated permissions: These services will allow a user to retrieve only the data of customers that their credential (as represented by the OAuth token) has access to. If an account or its data is targeted by a requestor but the user does not have permission, an error will be returned. This access will depend on the delegation permissions set up in myIR.

Gateway services like these typically have a 60 second timeout configured, although this may be adjusted after testing.

NOTE: For updates to versions of the SOAP architecture including the communication standards, security and service end points please follow the links provided in [section 4](#).

3 Operations

The schemas and WSDLs listed here are subject to change.

For the authoritative definitions, please visit

www.ird.govt.nz/digital-service-providers/services-catalogue

The structures of all Gateway Service operations are intended to produce the most efficient requests and responses. Any common structures and fields will be used across many schemas and tax types through an intentional inheritance method. The section below describes the structure of each operation and the scenarios in which certain fields will be used in XML requests and responses.

This section contains schema aliases:

- Cmn: Common.v1.xsd
- Sft: SoftwareIntermediation.v1.xsd

All requests and responses live in the SoftwareIntermediation.xsd.

All operations for the Software Intermediation Service will contain two standard header fields: **softwareProviderData** and **identifier**. The identifier value type will contain either "IRD", "ACCIRD" or "CST". This parameter will be different for each operation, as defined below.

The **identifier** field is common across all Gateway Services but refers to different parties in different services. In all cases it is the party with delegated permissions to whom an OAuth token is provided. If the value cannot be resolved to a known context, or if it can but the provided OAuth token does not have the necessary delegated permissions then the error code 4 "unauthorised delegation" is returned. Please refer to individual operations for the nature of the identifier expected in this parameter in any given context.

For example:

```
<cmn:softwareProviderData>
  <cmn:softwareProvider>SoftwareProvider</cmn:softwareProvider>
  <cmn:softwarePlatform>SoftwarePlatform</cmn:softwarePlatform>
  <cmn:softwareRelease>v1</cmn:softwareRelease>
</cmn:softwareProviderData>
<cmn:identifier IdentifierValueType="IRD">0123456789</cmn:identifier>
```

Field	Description
softwareProvider	The company that developed the software
softwarePlatform	The field value will be provided by Inland Revenue during the onboarding process
softwareRelease	The version of the software package

Field	Description
IdentifierValueType	The ID type being submitted which will be one of the following (see operations below for proper use): <ul style="list-style-type: none"> • IRD—IRD number • CLTLID—Client List Identifier • LSTID—List Identifier • ACCIRD—Account-specific IRD number
identifier	The value submitted for this field should contain only digits, with no dashes. IRD numbers that are eight digits must be padded with a leading zero.

Proper use:

- The only SoftwareProvider Data values that will be accepted are the ones that were provided to Inland Revenue at the time of onboarding.

The response structure for all requests will use the two default service response fields:

- **statusCode**
- **errorMessage**.

These will be the only two fields returned for the link/delink responses. For the retrieveList, these two fields will be returned as well as the requested client list.

For example:

```
<linkDelinkResponse xmlns="urn:www.ird.govt.nz/GWS:types/ReturnCommon.v1">
  <StatusMessage xmlns="urn:www.ird.govt.nz/GWS:types/Common.v1">
    <statusCode>0</statusCode>
    <errorMessage></errorMessage>
  </StatusMessage>
</linkDelinkResponse>
```

For a list of possible error codes and messages, see the '[Response codes](#)' section of this document.

3.1 Link

The Link operation will be used to link a software package to a customer or business intermediary client list. This operation will be called once per relationship upon first use. The request for this operation is defined in the SoftwareIntermediation schema and is called **LinkRequest**.

The Identifier field for the Link operation will have the IdentifierValueTypes of either "ACCIRD" or "IRD". This value indicates the party to be linked to the software platform. This party will be either of the following:

Intermediary type	ID type	ID type description	IRD format	Example
Bookkeeper	CLTLID	Client List Identifier	No	1047030
Other Representative	CLTLID	Client List Identifier	No	1022001
Payroll Bureau	CLTLID	Client List Identifier	No	1014002
Payroll Intermediary	IRD	IRD Number (of client list)	Yes	077415807
Payroll Intermediary	LSTID	List Identifier	Yes	077415807
Tax Agent	IRD	IRD Number (of client list)	Yes	123133609
Tax Agent	LSTID	List Identifier	Yes	123133609
Customer account**	ACCIRD	Account IRD number	Yes	019515842

** AccountType is required

The provided OAuth Token needs to have delegated permissions to this party.

Base structure:

```

<linkRequest
  xmlns="urn:www.ird.govt.nz/GWS:types/SoftwareIntermediation"
  xmlns:cmn="urn:www.ird.govt.nz/GWS:types/Common.v1"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="urn:www.ird.govt.nz/GWS:types/SoftwareIntermediation">
  <cmn:softwareProviderData>
    <cmn:softwareProvider>SoftwareProvider</cmn:softwareProvider>
    <cmn:softwarePlatform>SoftwarePlatform</cmn:softwarePlatform>
    <cmn:softwareRelease>v1</cmn:softwareRelease>
  </cmn:softwareProviderData>
  <cmn:identifier IdentifierValueType="ACCIRD">0123456789</cmn:identifier>
  <cmn:accountType>GST</cmn:accountType>
  <softwarePlatformID>1500068629</softwarePlatformID>
</linkRequest>
  
```

Field	Requirement	Description
identifier	Required	This value indicates the party to be linked to the software platform. The value submitted for this field should contain only digits, with no dashes. IRD numbers that are eight digits must be padded with a leading zero.
AccountType	Depending on context this field is required (for CUSTOMER) or ignored (for TAX AGENT).	The account type in this payload will specify the account type to which to link. This will only be done for links to CUSTOMERs and not AGENTs. (For distinction between the two see the RetrieveList operation)
SoftwarePlatformID	Required	This is the value provided by Inland Revenue during the registration process. This type of identifier is called a customer ID (a generic technical key for external parties in the Inland Revenue system) and is unique for every software package. It will be compared to the submitted software platform field.

NOTE: When this operation is being called, the *business intermediary or customer account* using the software **must** have owner or administrator access. The delegations are determined by the access currently granted in myIR.

The Link operation will not allow the same link to be created twice. If there is any uncertainty that a link exists there is no harm in calling the operation again, just ensure the first Link call has had time to process. The generic response fragment documented above is the only response since the processing happens asynchronously.

3.2 Delink

The Delink operation will be used to delink a customer or business intermediary client list from the software package. This operation will be called once per link upon final use. The removal of a link should only occur upon cessation of a relationship, while the renewal of a subscription does not require this operation.

The Identifier field for the Delink operation will have the IdentifierValueTypes of either the business intermediary or software package depending on who is submitting the Delink call. If the token is for someone with permissions on the software package then the identifier value will contain the software platform ID (of type customer ID so IdentifierValueType="CST"), and the targetID field will contain the agent/account to be delinked. If the token is for a person with permissions on a Tax Agent or Customer Account then this field will contain the corresponding ID and the targetId will contain the platform ID (CST).

The provided OAuth Token needs to have delegated permissions on this party.

Base structure:

```
<delinkRequest
  xmlns="urn:www.ird.govt.nz/GWS:types/SoftwareIntermediation"
  xmlns:cmn="urn:www.ird.govt.nz/GWS:types/Common.v1"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="urn:www.ird.govt.nz/GWS:types/SoftwareIntermediation">
  <cmn:softwareProviderData>
    <cmn:softwareProvider>SoftwareProvider</cmn:softwareProvider>
    <cmn:softwarePlatform>SoftwarePlatform</cmn:softwarePlatform>
    <cmn:softwareRelease>v1</cmn:softwareRelease>
  </cmn:softwareProviderData>

  <!-- Either -->
  <cmn:identifier IdentifierValueType="CST">1500123456</cmn:identifier>
  <targetId>123123123</targetId>
  <targetAccount>GST</targetAccount>

  <!-- OR -->
  <cmn:identifier IdentifierValueType="IRD">123123123</cmn:identifier>
  <targetId>1500123456</targetId>
  <targetAccount>GST</targetAccount>

</delinkRequest>
```

Field	Description
identifier	<ul style="list-style-type: none"> If the token resolves to an owner with permissions on the <i>software package</i> then the identifier value will contain the software platform ID (of type customer ID so IdentifierValueType="CST"), and the targetID field must contain the IRD number of the Tax Agent/Customer Account to be delinked. If the token resolves to a <i>business intermediary or customer account</i> with permissions on a Tax Agent or Customer Account then

Field	Description
	<p>this field will contain the corresponding ID and IdentifierValueType must be a valid ID (CLTLID, IRD, LISTID or ACCIRD), and the targetId must contain the platform ID.</p> <p>The value submitted for this field should contain only digits, with no dashes. IRD numbers that are eight digits must be padded with a leading zero.</p>
AccountType	Not used
targetId	<p>This field will be the other party to the link not already provided in the identifier.</p> <ul style="list-style-type: none"> • If the identifier is for the <i>software package</i> then the targetId will be that of a business intermediary or customer account • If the identifier is that of the <i>business intermediary or customer account</i> then the targetId will be the CST of the software package.
targetAccount	<p>This field will be used when the link being removed is an account-level link. If this field is not included in the payload and a customer-level link could not be found, account-level links will not be searched for. If the link was to a tax agent then this field will be ignored.</p>

Proper use:

There are two ways that this payload can be submitted, via the *software package* or the *business intermediary or customer account*. If the software package is requesting to delink, the software can submit this request one of two ways:

- A payload with an ID of the business intermediary or customer account (see table in section 3.1) as the **IdentifierType** and the CST identifier for the platform as the **targetID**.
- Alternatively, the request can be made later by the software package with the value of "CST" type as the **IdentifierType** and the *business intermediary or customer account* as the targetID.

The delinking of an account-level link requires the TargetAccount field to be populated. This means that any client with multiple account-level links will require multiple Delink calls for the different links.

NOTE: For special cases the Delink operation can be used to reissue bulk feed files. This operation will be called followed by the link operation. There is a short processing time required for the Delink operation to finish before the Link operation can be called. The generic response fragment documented above is the only response since the processing happens asynchronously.

3.3 RetrieveClientList

The RetrieveClientList operation will be used to retrieve all purchasers of a software package subscribed for bulk updates. This will only return active links and will not return delinked or expired relationships.

Base request structure:

```
<RetrieveListRequest
  xmlns="urn:www.ird.govt.nz/GWS:types/SoftwareIntermediation"
  xmlns:cmn="urn:www.ird.govt.nz/GWS:types/Common.v1"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="urn:www.ird.govt.nz/GWS:types/SoftwareIntermediation">
  <cmn:softwareProviderData>
    <cmn:softwareProvider>SoftwareProvider</cmn:softwareProvider>
    <cmn:softwarePlatform>SoftwarePlatform</cmn:softwarePlatform>
    <cmn:softwareRelease>v1</cmn:softwareRelease>
  </cmn:softwareProviderData>
  <cmn:identifier IdentifierValueType="CST">0123456789</cmn:identifier>
  <clientType>CUSTOMER</clientType>
  <clientAccountType>GST</clientAccountType>
</RetrieveListRequest>
```

Field	Description
identifier	Indicates the software platform. If this party cannot be resolved as a valid party, or if it can but the OAuth token is for someone who does not have permissions, in both cases a '4 unauthorised delegation' status code will be returned. The value submitted for this field should contain only digits, with no dashes. IRD numbers that are eight digits must be padded with a leading zero.
AccountType	Not used
clientType	This type is to distinguish between AGENT lists and CUSTOMER lists. AGENT lists will return the links to tax agents and CUSTOMER lists will return the links directly to customer accounts.
clientAccountType	Optional: This field will allow filtering by account type when clientType above is CUSTOMER.

Base response structure:

```
<retrieveListResponse
  xmlns="urn:www.ird.govt.nz/GWS:types/SoftwareIntermediation"
  xmlns:cmn="urn:www.ird.govt.nz/GWS:types/Common.v1"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="urn:www.ird.govt.nz/GWS:types/SoftwareIntermediation">
  <cmn:statusMessage>
    <cmn:statusCode>0</cmn:statusCode>
    <cmn:errorMessage/>
  </cmn:statusMessage>
```



```

<softwareUser>
  <client>
    <clientType>AGENT</clientType>
    <idType>IRD</idType>
    <id>123158616</id>
  </client>
  <client>
    <clientType>AGENT</clientType>
    <idType>LSTID</idType>
    <id>123744705</id>
  </client>
  <client>
    <clientType>AGENT</clientType>
    <idType>CLTLID</idType>
    <id>1035033</id>
  </client>

  <!-- OR -->

  <client>
    <clientType>CUST</clientType>
    <idType>ACCIRD</idType>
    <id>123089456</id>
    <accountType>GST</accountType>
  </client>
  <client>
    <clientType>CUST</clientType>
    <idType>ACCIRD</idType>
    <id>123089456</id>
    <accountType>INC</accountType>
  </client>
</softwareUser>
</retrieveListResponse>
  
```

Field	Description
clientType	Depending on the request, only a list of AGENTs or only a list of CUSTOMERs will be returned
idType	This value will either be an Account IRD (ACCIRD) or IRD. Tax agencies will be the only ones with IRD as the identifier.
Id	This is the value based on the idType
accountType	This value only applies to ACCIRD idTypes

4 End points, schemas and WSDLs

Current environment information for this service—including the end points for each environment, schemas and WSDLs—is available within the relevant Software Development Kit (SDK).

To access the SDK, do one of the following:

- Go to <https://github.com/InlandRevenue> and select this service
- Go to <https://developerportal.ird.govt.nz> and click the link to the SDK within the Gateway Service documentation (please register first).

4.1 End points

See instructions above for where to find end points for this service.

4.2 Schemas

All schemas for the Software Intermediation Service import a common.xsd which has some data types specific to Inland Revenue. This common.v1.xsd will be used in other Gateway Services outside of the /SoftwareIntermediation/ namespace so it must be kept up-to-date, without numerous redundant versions remaining. The schemas for all operations will import SoftwareIntermediation.xsd for the request and response.

See instructions at beginning of section 4 for where to find schemas for this service.

4.3 WSDLs

The Software Intermediation Gateway Service has one WSDL, which has a target namespace of <https://services.ird.govt.nz/GWS/SoftwareIntermediation> and can be found at <https://services.ird.govt.nz:4046/gateway/GWS/SoftwareIntermediation/?singleWsd>

Note: The production URL above will not work until you have onboarded with Inland Revenue.

As explained in the '[Solution design](#)' section of this document, the online WSDLs have 'any' elements underneath the wrapper elements. There is a development version of the WSDL available with this build pack that replaces the 'any' element with an imported reference to the schema to facilitate initial development and testing. To consume the actual service, the binding will need to be done at the hosted end point. However, for initial development this static WSDL can be used.

All WSDL messages follow this naming convention:

```
SoftwareIntermdiation_<operation>_InputMessage

<wsdl:portType name="SoftwareIntermediation">
  <wsdl:operation name="Link">
  <wsdl:operation name="Delink">
  <wsdl:operation name="RetrieveList">
  <wsdl:service name="SoftwareIntermediation">
```

See beginning of [section 4](#) for instructions on where to find WSDLs for this service.

5 Response codes

The response message from the Gateway Services always includes a status code and status message that describes how successfully the gateway service call was carried out. Following the status message will be the responseBody, which will return the operations response.

5.1 Generic gateway response codes

The following response codes are common to all Gateway Service calls. The operations for the Software Intermediation Service all apply customer-level security validation at the framework level and the descriptions for these codes reflect that.

Standard codes	Standard message	Description
-1	An unknown error has occurred	This is generally what will be returned for internal errors that are not due to the service request
0	Success	Standard success code is 0
1	Authentication failure	General authentication failure status
2	Missing authentication token(s)	Tokens were not included in the HTTP header as expected
3	Unauthorised access	Access is not permitted for the requester to use the gateway services. Access could not be confirmed due to OAuth token validation failing. This could be due to invalid format of the token or infrastructure being unavailable.
4	Unauthorised delegation	<p>Access is not permitted for the requester to perform this operation for the submitted identifier. This code will be returned in any of these situations:</p> <ul style="list-style-type: none"> The submitted cmn:identifier has an invalid value. The identifier type (IdentifierValueType attribute on cmn:identifier) supplied is invalid. All the values above are valid but the provided OAuth token does not have delegated access to that Customer. The Delink operation will NOT return this code but will return 113 as per below.
5	Unauthorised vendor	Vendor is not permitted access—has not been onboarded for this operation
20	Unrecognised XML request	This could be due to the external sender sending in incorrect XML or it could be due to bad/poor/missing configuration

Standard codes	Standard message	Description
21	XML request failed validation	The external requestor submitted XML that is not formatted according to our defined schemas
(none)	(non XML)	In some scenarios where the request message does not have a well formed XML structure or is not valid or does not adhere to the SOAP protocol formats, the framework generates a parsing exception that is not wrapped in XML nor has a response status code
(none)	(SOAP fault) UnAuthorised	When maximum concurrency has been exceeded by the service provider this SOAP fault will be returned

5.2 Generic software intermediation response codes

The following response codes are specific to Software Intermediation Service calls:

Standard codes	Standard message	Description	Link	Delink	Retrieve List
100	Could not extract data from xml payload	Could not extract data from XML payload	X	X	X
101	Link already exists	A link already exists between the parties attempting to link	X		
102	No active link	A link does not exist between the parties attempting to delink		X	
103	Unable to save request	A request was not created. Submit request again.	X	X	
104	Missing account type	An account IRD number was submitted without an account type		X	
105	Cannot create customer level link	An IRD number was submitted without an account type and the IRD number does not belong to a tax agency	X		
106	Missing target account	A target account type was not provided to find the account from which to delink		X	
108	No account IRD provided for account	No account IRD number (ACCIRD) provided for the given account type	X		
109	The specified software has no links of that type	The specified software intermediary has no links of that type. Link types are either AGENT or CUSTOMER.			X

Standard codes	Standard message	Description	Link	Delink	Retrieve List
111	No software registration found	No software intermediary found for the given Software Platform Customer ID	X	X	
112	No agent or customer found	No agent or customer found with provided identifier		X	
113	No account found	No account found for the given account type		X	
114	The provided account type is not supported	The provided account type is not authorised in the current operation	X		

6 Glossary

Acronym/term	Definition
ACCIRD	Account IRD number
Authentication	The process that verifies the identity of the party attempting to access Inland Revenue
Authorisation	The process of determining whether a party is entitled to perform the function or access a resource
Build pack	Details the technical requirements and specifications, processes and sample payloads for the specified activity
Client	As used in this build pack client generally refers to the party licensing and using the software intermediary/software provider's software
Credentials	Information used to authenticate identity, for instance an account username and password
CST	Customer identifier
Customer	<p>A customer is the party who is a tax payer or a participant in the social policy products that are operated by Inland Revenue. The customer might be a person (an "individual") or a non-individual entity such as a company, trust, society etc.</p> <p>Practically all of the service interactions with Inland Revenue are about a customer (such as their returns, accounts, entitlements etc) even though these interactions might be undertaken by an Intermediary such as a tax agent on their behalf.</p>
Encryption	Cryptographic transformation of data (called "plaintext") into a form (called "cipher text") that conceals the data's original meaning to prevent it from being known or used. If the transformation is reversible, the corresponding reversal process is called "decryption", which is a transformation that restores encrypted data to its original state (see RFC 2828).
End points	A term used to describe a web service that has been implemented
GWS	Gateway Services—the brand name for the suite of web services that Inland Revenue is providing. The Software intermediation Service is a Gateway Service.
HTTP, HTTPS	Hyper Text Transmission Protocol (Secure)—the protocol by which web browsers and servers interact with each other. When implemented over TLS1.2 HTTP becomes HTTPS.
IP	Internet Protocol—the principal communication protocol in the Internet protocol suite for relaying datagrams across networks
IRD	Inland Revenue Department (ie IRD number)
NZISM	NZ Information Security Manual—the security standards and best practices for Government agencies. Maintained by the NZ Government Communications Security Bureau (GCSB).
OAuth 2.0	OAuth 2.0 is an industry-standard protocol for authorisation
Pattern	A constraint on data type values that require the string literal used in

Acronym/term	Definition
	the data type's lexical space to match a specific pattern
Payloads	This refers to the data contained within the messages that are exchanged when a web service is invoked. Messages consist of a header and a payload.
Schemas	An XML schema defines the syntax of an XML document, in particular of a payload. The schema specifies what a valid payload (such as a GST return) must/can contain, as well as validating the payload.
SHA	Secure Hashing Algorithm. There is a family of these that provide different strengths. SHA-2 is currently favoured over SHA-1, which has been compromised.
Software provider	The organisation developing the software connecting to Inland Revenue gateway services—also known as software intermediary, software developer or service provider
Software provider software	<p>A client application is an operating instance of software that is deployed in one or more sites. A number of deployment patterns are possible:</p> <ol style="list-style-type: none"> 1. A single cloud based instance with multiple tenants and online users 2. An on premise instance (eg an organisation's payroll system) 3. A desktop application with an online user. <p>This is the computer software that contains interfaces to consume the services that Inland Revenue exposes. Software is developed and maintained by a software developer and subsequently deployed as one or more client applications.</p>
SFTP	Secure File Transport Protocol. SFTP 3.0 is used.
Solution	The technology components, systems and interface specifications constituting the Tax Agent Web Services capability which enables integration and communication across the Gateway channel between Inland Revenue and tax agents for the purpose of providing the service
SOAP	Simple Object Access Protocol—a set of standards for specifying web services. GWS uses SOAP version 1.2
SSL	Secure Sockets Layer certificates—used to establish an encrypted connection between a browser or user's computer and a service or website
START	Simplified Taxation and Revenue Technology—Inland Revenue's new core tax processing application. It is an implementation of the GenTax product from FAST Enterprises.
Tax agent	A tax agent who is formally registered as such with Inland Revenue
TDS	Transaction Data Services
TLS1.2	Transport Layer Security version 1.2—the protocol that is observed between adjacent servers for encrypting the data that they exchange. Prior versions of TLS and all versions of SSL have been compromised and are superseded by TLS1.2.

Acronym/term	Definition
URL	Universal Resource Locator—also known as a web address
User	The user referred to in this document is the user of the software provider accounting or tax package. This user needs delegated permissions on customer tax accounts (potentially via a tax agency or other intermediary) in order to use TDS. The web logon used in eServices needs to be used in making Inland Revenue queries. This web logon must be granted permission there to access customer accounts.
WSDL	Web Service Definition Language—an XML definition of a web service interface
X.509 certificate	An international standard for encoding and describing a digital certificate. In isolation a public key is just a very large number, the X509 certificate to which it is bound identifies whose key it is, who issued it, when it expires etc. When a counterparty's X509 digital certificate is received, the recipient takes their public key out of it and store the key in their own keystore. The recipient can then use this key to encrypt and sign the messages that they exchange with this counterparty.
XIAMS	External IAMS—an instance of IAMS that authenticates and authorises access by external parties, for example customers or trading partners, as opposed to internal parties such as staff
XML	Extensible Mark-up Language—a language used to define a set of rules used for encoding documents in a format that can be read by humans and machines
XSD	XML Schema Definition—the current standard schema language for all XML data and documents

7 Change log

This table lists all changes that have been made to this build pack document since the release of version 1.

Version	Date of change	Document section	Description
	24/08/20	3.3	<ul style="list-style-type: none"> Revised example of client list response
		1.3	<ul style="list-style-type: none"> Change TDS description from API to services
		3	<ul style="list-style-type: none"> List of ID types updated in description of IdentifierValueType (NOTE: supported ID types have been updated throughout document to include IRD, CLTLID, LSTID and ACCIRD)
		3.1	<ul style="list-style-type: none"> Revise the link operation description
			<ul style="list-style-type: none"> Added table of supported ID types
			<ul style="list-style-type: none"> Remove "CST" from supported IdentifierValueType and added list of supported types
		3.2	<ul style="list-style-type: none"> Revised field descriptions
			<ul style="list-style-type: none"> Changed customer to business intermediary or customer account
		1.3	<ul style="list-style-type: none"> Section name changed from 'Related build packs' to 'Related services' and text modified.
		1.4	<ul style="list-style-type: none"> Note added to Prerequisites table: Note that the same certificate cannot be used for the Test and Production environments.
		2.4	<ul style="list-style-type: none"> TLS information updated to reflect use of TLS1.3
		4.1	<ul style="list-style-type: none"> Removed end points and replaced with link: <i>Onboarding instructions are available at https://www.ird.govt.nz/software-providers/.</i>
		5	<ul style="list-style-type: none"> Error code 6 removed (authentication expired)
		1.1	<ul style="list-style-type: none"> Updates made to boxed instructions for where to find additional information such as business-level context, use cases and links to relevant policy.
		1.3	<ul style="list-style-type: none"> Updated instructions on where to find related build packs.
		2.4	Text added at end of section: <ul style="list-style-type: none"> <i>For updates to versions of the SOAP architecture including the communication standards, security and service end points please follow the links provided in section 4.</i>

Version	Date of change	Document section	Description
		3	<ul style="list-style-type: none"> Updated hyperlink in boxed text at start of section
		4	<ul style="list-style-type: none"> Removed boxed instructions on where to find current end points, schemas and WSDLs and updated with new instructions.
		4.1	Text updated to this: <ul style="list-style-type: none"> <i>See instructions above for where to find end points for this service.</i>
		4.2	Note added: <ul style="list-style-type: none"> <i>See instructions at beginning of section 4 for where to find schemas for this service.</i>
		4.3	Removed redundant note at end of section regarding WSDLs. Added following text: <ul style="list-style-type: none"> <i>See beginning of section 4 for instructions on where to find WSDLs for this service.</i>
	05/03/19	3	<ul style="list-style-type: none"> Description of softwarePlatform in table removed: <i>"The software package that is making the request".</i> Replaced with new description: <i>"The field value will be provided by Inland Revenue during the onboarding process."</i>
	03/12/18	(formerly section 6)	<ul style="list-style-type: none"> REMOVED: Section on use cases and scenarios. Readers are instead advised to visit https://www.ird.govt.nz/software-providers/ for such information
		4.3	<ul style="list-style-type: none"> Removed: The Software Intermediation Gateway Service has one WSDL, which has a target namespace of https://services.ird.govt.nz/GWS/SoftwareIntermediation/ and can be found at https://services.ird.govt.nz/GWS/SoftwareIntermediation/?singleWsdL. <p>Replaced with: <i>The Software Intermediation Gateway Service has one WSDL, which has a target namespace of https://services.ird.govt.nz/GWS/SoftwareIntermediation/ and can be found at https://services.ird.govt.nz:4046/gateway/GWS/SoftwareIntermediation/?singleWsdL</i></p>

Version	Date of change	Document section	Description
			<p>Note: The production URL above will not work until you have onboarded with Inland Revenue.</p> <ul style="list-style-type: none"> Removed: Note: The WSDL hosted at the end point above will not contain these XSD references, only the static WSDL provided on our GitHub page. Replaced with: Note: The WSDL hosted at the end point above will not contain these XSD references, only the static WSDL provided at https://www.ird.govt.nz/software-providers/.
		3.1	<ul style="list-style-type: none"> Table reformatted to include column for 'requirement'
		3 4	<ul style="list-style-type: none"> Removed: IMPORTANT: The end points, schemas and WSDLs listed here are subject to change. For the authoritative definitions, please refer to the information provided on the Inland Revenue Gateway Services GitHub site: https://github.com/InlandRevenue/Gateway-Services Replaced with: IMPORTANT The end points, schemas and WSDLs listed here are subject to change. For the authoritative definitions, please visit https://www.ird.govt.nz/software-providers/
		Table of contents	<ul style="list-style-type: none"> Removed 'List of tables' and 'List of figures'
		Entire document	<ul style="list-style-type: none"> Removed all 'table' and 'figure' captions
		1.4	<ul style="list-style-type: none"> Removed row from prerequisites table on Inland Revenue: INLAND REVENUE: Provide the Inland Revenue public certificate for mutual TLS. Inland Revenue's public X.509 certificate to support TLS will be provided as part of connectivity testing.
		1.1	<ul style="list-style-type: none"> REMOVED: The associated onboarding and overview documents describe the end-to-end business level solution, of which this build pack forms part. Development versions of schemas, and sample requests and responses are also

Version	Date of change	Document section	Description
			<p>available with this build pack.</p> <ul style="list-style-type: none"> REMOVED: section called 'Onboarding packs for supported processes' ADDED: <p>Before you continue, please be sure to consult http://www.ird.govt.nz/software-providers/ for the products that use this Service, business-level context and use cases, links to relevant policy, and information on how to integrate with Inland Revenue's products and services.</p>
		Glossary	<ul style="list-style-type: none"> Entries added to glossary
	19/11/18	1.4.1	<ul style="list-style-type: none"> Section 1.4.1 added to cover mutual TLS security
		7 Glossary 8 Change log	<ul style="list-style-type: none"> Removed word 'Appendix' from both headings
1.00	13/04/18	Entire doc	<ul style="list-style-type: none"> V1