Inland Revenue
Te Tari Taake

Inland Revenue

Build Pack: Return Service—Accounting Income Method Version 2.0

Date: 29/06/2021

Inland Revenue
Te Tari Taake

Contents

Inland Revenue
Te Tari Taake

1 Overview

1.1 This solution

Inland Revenue has a suite of digital services available for consumption by our service providers that support efficient, electronic business interactions with Inland Revenue. The Accounting Income Method (AIM) Return Service described in this build pack document forms part of a suite of Gateway Services.

This is a stand-alone document intended to provide the technical details required to support the end-to-end onboarding Gateway Services. It describes the architecture of the technical solution, schemas, end points, sample payloads to use in non-production environments, 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 AIM Return service. The reader is assumed to have a suitable level of technical knowledge to comprehend the information provided.

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 Identity and Access Services (required)

The 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 Return Service.

1.3.2 Intermediation Service (recommended)

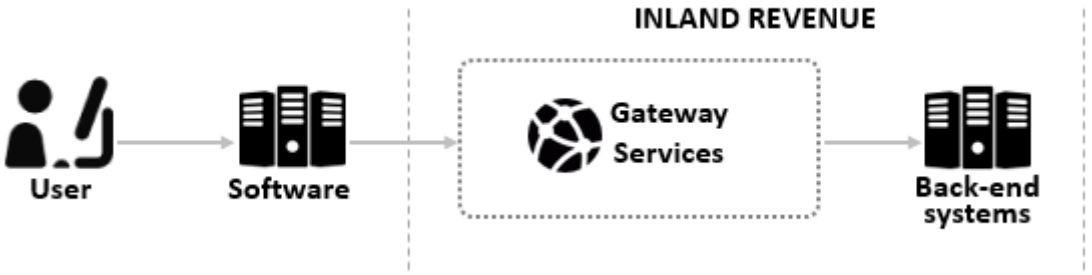
The Intermediation Service build pack supports software providers with the process of linking tax intermediaries (such as tax agents) to their clients so intermediaries can act on their behalf through the Return service.

2 Solution design

2.1 Architecture

Inland Revenue's Gateway Services suite is used by approved service providers to facilitate everything from registration activities, filing returns, making payments and other service offerings to allow customers to interact with Inland Revenue.

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



The WSDLs for the Return Service 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.

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

2.2 Service scope

The Return Service supports the following operations:

- **File:** This operation is used to submit a return to Inland Revenue for a customer.
- **Prepop:** This operation is used by software to provide figures to assist in the calculation and display of return information prior to submission.
- **RetrieveStatus:** This operation is used by software to return a status for a particular return.
- **RetrieveReturn:** This operation retrieves a previously submitted return and the values associated with that return.
- **RetrieveFilingObligation:** This operation retrieves the expectations of a customer to file a return.

2.3 Messaging

The Return Service is a SOAP-based web service. All SOAP messages require a SOAP header and a SOAP body containing a structured XML payload. Correct values can be found in the relevant WSDL, the link to which is provided in section 4 of this document.

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 schema that define the requests.

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

Note that the Gateway Services use the SOAP version 1.2 protocol, and the SOAP service contract is published using WSDL version 1.1.

Example SOAP request structure

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
        xmlns:ret="https://services.ird.govt.nz/GWS/Returns/"
        xmlns:prep="https://services.ird.govt.nz/GWS/Returns/:types/PrepopRequest"
        xmlns:a="http://www.w3.org/2005/08/addressing">
    <soap:Header>
        <a:Action>https://services.ird.govt.nz/GWS/Returns/Return/Operation</a:Action>
    </soap:Header>
    <soap:Body>
        <ret:Prepop>
            <ret:ReturnPrepopRequestMsg>
                <prep:PrepopRequestWrapper>
                    <rc:retrieveFormInfoRequest xmlns:xsi…
                        <…PrepopFields…>
                    </rc:retrieveFormInfoRequest>
                </prep:PrepopRequestWrapper>
            </ret:ReturnPrepopRequestMsg>
        </ret:Prepop>
    </soap:Body>
</soap:Envelope>
```

Inland Revenue
Te Tari Taake

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">
    <s:Header>
        <a:Action s:mustUnderstand="1">
        https://services.ird.govt.nz/GWS/Returns/Return/FileResponse
        </a:Action>
    </s:Header>
    <s:Body>
        <FileResponse xmlns="https://services.ird.govt.nz/GWS/Returns/">
        <FileResult xmlns:b=https://services.ird.govt.nz/GWS/Returns/:types/FileResponse
        xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
            <b:FileResponseWrapper>
                <fileResponse xmlns="urn:www.ird.govt.nz/GWS:types/Common.v2">
                  <statusMessage>
                     <statusCode>0</statusCode>
                     <errorMessage/>
                  </statusMessage>
                </fileResponse>
             </b:FileResponseWrapper>
          </FileResult>
        </FileResponse>
    </s:Body>
</s:Envelope>
```

2.4 Security

2.4.1 Information classification

The information exchanged via the Return Service has an information classification of "**IN CONFIDENCE**". The following security standards therefore apply.

2.4.2 Transport Layer Security and certificates

Mutual Transport Layer Security (TLS) is implemented for this service. This requires the use of a publicly-issued X.509 certificate from one of the trusted certificate authorities listed further below in this section. (Note that Inland Revenue does not issue certificates to external vendors for web service security implementations.)

Inland Revenue has the following requirements for accepting public X.509 keys:

- ECDSA (preferred) key length: 384 bits (or RSA key length: 2048 bits)
- Self-signed certificates are not accepted
- Certificates issued by private/internal certificate authorities are not accepted
- The same certificate cannot be used for the Test and Production environments.

Inland Revenue has adopted a trust-based authentication model and will only accept certificates that contain a pre-approved subject common name and have been issued by one of the following root certificate authorities, trusted and approved by Inland Revenue:

- Amazon
- Comodo
- DigiCert
- Entrust
- GeoTrust
- Let's Encrypt
- Sectigo
- Thawte.

Inland Revenue expects Digital Service Providers to use their Inland Revenue Developer Portal account to create their common name for both Test and Production certificates. Please refer to the Digital Service Providers pages on the Inland Revenue website or contact your Inland Revenue onboarding representative at GatewayServices@ird.govt.nz for further details.

2.4.3 Ciphers

While Inland Revenue currently supports TSL1.2, it is migrating to TLS1.3 which specifies a much smaller and more prescriptive suite of ciphers. As Inland Revenue's security gateways do not currently support the CCM mode (*counter with cipher block chaining message authentication code*) of operation, only the following ciphers will be supported over TLS1.3:

Status	TLS1.3 ciphers
Supported now and in the future	• TLS_AES_128_GCM_SHA256 • TLS_AES_256_GCM_SHA384 • TLS_CHACHA20_POLY1305_SHA256

The following TLS1.2 ciphers are currently supported but some will be deprecated as below:

Status	TLS1.2 ciphers
Supported now and in future	• TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 • TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
Supported now but will be deprecated on 31 March 2022	• TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA • TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA • TLS_RSA_WITH_AES_128_CBC_SHA • TLS_RSA_WITH_AES_256_CBC_SHA • TLS_DHE_RSA_WITH_AES_128_CBC_SHA • TLS_DHE_RSA_WITH_AES_128_CBC_SHA256 • TLS_DHE_RSA_WITH_AES_256_CBC_SHA • TLS_DHE_RSA_WITH_AES_256_CBC_SHA256 • TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 • TLS_DHE_RSA_WITH_AES_256_GCM_SHA384

Status	TLS1.2 ciphers
Supported now but will be deprecated on 31 December 2022	• TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 • TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 • TLS_RSA_WITH_AES_128_CBC_SHA256 • TLS_RSA_WITH_AES_256_CBC_SHA256 • TLS_RSA_WITH_AES_128_GCM_SHA256 • TLS_RSA_WITH_AES_256_GCM_SHA384

2.4.4 End points

There are 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 service providers' centralised **cloud** locations can connect. This will require X.509 certificates for mutual TLS with an agreed common name, however certificates no longer need to be exchanged with Inland Revenue. On the cloud end point, Inland Revenue has controls to shield service providers from issues caused by heavy usage from other providers.

2. For service providers connecting from **desktops/native apps** that are unable to securely store certificates and access tokens. There is a separate end point that does not use mutual TLS and therefore does not require certificates. On the desktop end point, Inland Revenue has less ability to shield consumers of the service from heavy usage by others.

	End point for cloud-based connections	End point for desktop connections
Purpose	• Primary preferred end point to connect to from service providers for Gateway Services	• Additional transitory 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 set up certificate trust
Client application type	• Cloud applications	• Desktop/native applications • For connecting from multiple decentralised clients
Constraints	• Only for source locations with client-side TLS certificates • On the cloud end point Inland Revenue has controls to shield service providers from issues caused by heavy usage from other providers	• 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

Inland Revenue
Te Tari Taake

	End point for cloud-based connections	End point for desktop connections
Mutual TLS	• Inland Revenue explicitly trusts the certificate the service provider associates with the TLS connection as client for Mutual TLS connections and uses it to identify the service provider in conjunction with the web service identification below	• Server-side certificates only
Minimum TLS version	• 1.2	• 1.2
URL	• Contains …/gateway/..	• Contains …/gateway2/..
Port	• 4046	• 443 (Default https port)
Web service consumer identification	• To be identified in web service calls—each cloud application will be given client_id/client_secret credentials during onboarding to allow it to call this end point	• Desktop clients will be given client_id/client_secret credentials in the same manner as cloud application clients. However, desktop clients will not be able to redeem refresh tokens to obtain a new OAuth token when it expires.
Firewalling in production	• No IP address restrictions • Access limited by certificate enrolment	• No IP address restrictions
Firewalling in non-production environments	• No IP address restrictions • Access limited by certificate enrolment	• Firewalled—IP whitelisting needed

Inland Revenue
Te Tari Taake

2.4.5 Authentication and authorisation

Authentication and authorisation are the mechanisms by which the consumer of the service is identified, and their access rights enforced. The Return Service uses the standard OAuth2 authorisation code flow. For instructions on how to acquire an OAuth access token, and the properties of this token (eg its expiry and refresh parameters) please refer to the Identity and Access build pack.

Authentication and authorisation are described in terms of two parties:

- **Consumer**—this is the party under whose identity the interaction is being transacted (the party who has been authenticated)
- **Resource**—this is the data entity/object being accessed (eg created, read, updated or deleted) via the service.

When using OAuth, the consumer is authenticated using their Inland Revenue myIR credentials and their access is authorised using the same access rights as myIR. For example, if a myIR 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 following steps are applied by the Gateway Services when authorising access by the consumer to a resource:

1. If the consumer is the resource owner then access to the resource is authorised (ie the consumer is authorised to manage their own affairs).
2. Otherwise, if the consumer's myIR credential has been granted access to the resource, with the appropriate level of access, then access is authorised.
3. Otherwise, if the consumer is an intermediary of an appropriate type who has been delegated access by being linked to the resource, with the appropriate level of access, then access is authorised.
4. Otherwise access is denied.

Inland Revenue
Te Tari Taake

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.v2
- rc: ReturnCommon.v2
- r: ReturnAIM.v2

NOTE: Some requests and responses live in ReturnCommon.v2.xsd but can still be generated from an inheriting return-specific XSD. This could mean the schemaLocation could be different, depending on where the payload was generated from. Any method of generating these payloads is accepted. This applies to the fileResponse XML directly below.

The response structure for all File requests will use the two default service response fields: **statusCode** and **errorMessage**. The identifier for this XML is fileResponse in the ReturnCommon namespace.

The response structure for all File requests will have the **gatewayId** field populated. The gatewayId is a unique identifier passed back in the responseBody, assuming the response code for the request is zero (refer to Chapter 5 Responses). The gatewayId should be recorded and can be used by technical teams for troubleshooting. The gatewayId will not appear in search results when searching myIR. The gatewayId is also not available to Inland Revenue front-line staff (such as in the telephone contact centre) to search on.

For example:

```
<fileResponse xmlns="urn:www.ird.govt.nz/GWS:types/ReturnCommon.v2">
        <StatusMessage xmlns="urn:www.ird.govt.nz/GWS:types/Common.v2">
            <statusCode>0</statusCode>
            <errorMessage></errorMessage>
        </StatusMessage>
        <responseBody>
             <gatewayId>0000 002J ZJ5N 6</gatewayId>
        </responseBody>
</fileResponse>
```

All operations for the Return service will contain two standard header fields: **softwareProviderData** and **identifier**.

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>v2</cmn:softwareRelease>
</cmn:softwareProviderData>
<cmn:identifier IdentifierValueType="ACCIRD">012345678</cmn:identifier>
<cmn:accountType>INC</cmn:accountType>
```

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
IdentifierValueType	The ID type being submitted. This can be ACCIRD, NZBN or ACC. The value submitted for this field should contain only digits, with no dashes (with the exception of the ACC field, which may contain letters to identify the account type). IRD Numbers that are eight digits must be padded with a leading zero.
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.
accountType	The account type being submitted (INC, IIT, ITN).

Proper use:

- The only softwareProviderData fields users will be able to input are the ones that were provided to Inland Revenue at the time of on-boarding.
- The identifier is that of the taxpayer on whose behalf the operations are being performed.

Example scenario:

- Third party with IRD 898989898 submits for client IRD 121212121
    - Third party calls /Returns/File/ with <cmn:identifier IdentifierValueType="ACCIRD">121212121</cmn:identifier>

3.1 File

The File operation will be used to submit AIM returns.

Base structure:

```
<r:fileRequest namespaces...>
      <rc:fileHeader>...</rc:fileHeader>
      <rc:fileBody>
         <rc:standardFields>
         <rc:formFields xsi:type="r:FormFieldsType">
                  <...tax specific fields...>
         </rc:formFields>
       </rc:fileBody>
 </r:fileRequest>
```

Field	Description
fileHeader	The standard header for File requests
fileBody	The standard body structure for File requests
standardFields	A group of standard fields
formFields	A wrapper that will contain tax form-specific fields

<**FileHeader**> structure:

```
<r:fileRequest namespaces...>
    <rc:fileHeader>
        <cmn:softwareProviderData>
                <cmn:softwareProvider>SoftwareProvider/cmn:softwareProvider>
                <cmn:softwarePlatform>SoftwarePlatform</cmn:softwarePlatform>
                <cmn:softwareRelease>v2</cmn:softwareRelease>
        </cmn:softwareProviderData>
        <cmn:identifier IdentifierValueType="ACCIRD">012345678</cmn:identifier>
        <cmn:accountType>INC</cmn:accountType>
        <rc:periodEndDate>2019-04-30</rc:periodEndDate>
        <rc:majorFormType>SOA</rc:majorFormType>
        <rc:minorFormType>SOA_2</rc:minorFormType>
   </rc:fileHeader>
     <rc:fileBody>
        <rc:standardFields>
        <rc:formFields xsi:type="r:FormFieldsType">
                <...tax specific fields...>
        </rc:formFields>
      </rc:fileBody>
 </r:fileRequest>
```

Field	Requirement	Description
periodEndDate	Required	The period in which a return exists or the period for which it is being submitted. An AIM Statement of Activity will cover a one or two-month period. The period end date for the Statement of Activity refers to the last day of the period covered by that statement. For example, for a Statement of Activity covering the two-month period of April and May 2019, the period end date is 31/05/19.
majorFormType	Required	The form type (SOA)
minorFormType	Required	The minor form type (SOA_2)

<**FileBody**> structure:

```
<r:fileRequest namespaces...>
        <rc:fileHeader>...</rc:fileHeader>
        <rc:fileBody>
                <rc:standardFields>
                <rc:formFields xsi:type="r:FormFieldsType">
                        <...tax specific fields...>
                </rc:formFields>
      </rc:fileBody>
 </r:fileRequest>
```

FileBody is simply the wrapper of standardFields and formFields. The standard fields will be constant in every fileBody, but the formFields will be overridden by each tax type.

<**StandardFields**> structure:

```
<r:fileRequest namespaces...>
     <rc:fileHeader>...</rc:fileHeader>
     <rc:fileBody>
        <rc:standardFields>
                <rc:isNilReturn>false</rc:isNilReturn>
                <rc:amendmentRequest>
                        <rc:isAmended>false</rc:isAmended>
                        <rc:amendReason></rc:amendReason>
                        <rc:amendDetails></rc:amendReason>
                </rc:amendmentRequest>
                <rc:creditTransferRequest>
                        <rc:transferIRD></rc:transferIRD>
                        <rc:transferAccountType></rc:transferAccountType>
                        <rc:transferFilingPeriod></rc:transferFilingPeriod>
                        <rc:associatedCustomer></rc:associatedCustomer>
                        <rc:transferAmount></rc:transferAmount>
                </rc:creditTransferRequest>
        </rc:standardFields>
        <rc:formFields xsi:type="r:FormFieldsType">
                <...tax specific fields...>
        </rc:formFields>
      </rc:fileBody>
 </r:fileRequest>
```

Field	Requirement	Description
isNilReturn	Required	This allows for a nil return to be submitted
isAmended	Required	This allows for a return to be filed as an amendment. **NOTE:** If isAmended=true then amendReason and amendDetails are required. Otherwise empty values are required in the amendReason and amendDetails fields.
amendReason	Conditional	This is attached to the amendmentRequest as the reason for the amendment. This can be either KEY (incorrect amount), MATH (calculation error), OTHER, or TRNSPO (transposition error).
amendDetails	Conditional	This allows for any further details on the amendmentRequest.
creditTransferRequest	Optional	These fields can be added to transfer the refund to another START account. Any number of credit transfers from 0 to 20 can be submitted for every file operation.

Proper uses:

- Most standard submissions will require isNilReturn to be *false* and isAmended to be *false.*

Example scenario:

- Attempting to amend an AIM return due to lack of information from client.
  <rc:isNilReturn>false</rc:isNilReturn>
  <rc:isAmended>true</rc:isAmended>
  <rc:amendReason>KEY</rc:amendReason>
  <rc:amendDetails>Client's previous months' income changed after initial filing</rc:amendDetails>

Inland Revenue
Te Tari Taake

<**FormFields**> structure:

```
<r:fileRequest xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:cmn="urn:www.ird.govt.nz/GWS:types/Common.v2"
        xmlns:r="urn:www.ird.govt.nz/GWS:types/ReturnAIM.v2"
        xmlns:rc="urn:www.ird.govt.nz/GWS:types/ReturnCommon.v2"
        xsi:schemaLocation="urn:www.ird.govt.nz/GWS:types/ReturnAIM.v2">
        <rc:fileHeader>…</rc:fileHeader>
        <rc:fileBody>
                <rc:standardFields/>
                <rc:formFields xsi:type="r:FormFieldsType">
                        <r:midYearEntry/>
                        <r:overFiveMillion/>
                        <r:aimInstalmentDate/>
                        <r:grossSalesAndServiceIncome/>
                        <r:openingStock/>
                        <r:purchases/>
                        <r:closingStock systemAdjustedValue="" userAdjustedValue=""/>
                        <r:grossProfit/>
                        <r:interestReceived/>
                        <r:dividendsReceived/>
                        <r:rentLeaseLicenceIncome/>
                        <r:otherIncome/>
                        <r:badDebts/>
                        <r:depreciationAndAmortisation/>
                        <r:insurance/>
                        <r:interestExpense/>
                        <r:fees/>
                        <r:rates/>
                        <r:rentsLeasesLicences/>
                        <r:repairsAndMaintenance/>
                        <r:researchAndDevelopment/>
                        <r:relatedPartyRemuneration/>
                        <r:salariesAndWages/>
                        <r:contractorPayments/>
                        <r:otherExpenses/>
                        <r:exceptionalItems/>
                        <r:netProfitLossBeforeTax/>
                        <r:taxAdjustments/>
                        <r:currentYearTaxableProfitLoss/>
                        <r:accountsReceivable systemAdjustedValue="" userAdjustedValue=""/>
                        <r:cashAndDeposits/>
                        <r:otherCurrentAssets/>
                        <r:vehicles/>
                        <r:plantAndMachinery/>
                        <r:furnitureAndFittings/>
                        <r:land/>
                        <r:buildings/>
                        <r:otherFixedAssets/>
                        <r:intangibles/>
                        <r:sharesAndOwnershipInterests/>
                        <r:termDeposits/>
                        <r:otherNonCurrent/>
                        <r:provisions systemAdjustedValue="" userAdjustedValue=""/>
                        <r:provisionsForShareholderSalaries systemAdjustedValue=""
                userAdjustedValue=""/>
                        <r:accountsPayable systemAdjustedValue userAdjustedValue=""/>
                        <r:currentLoans/>
                        <r:otherCurrentLiabilities/>
                        <r:nonCurrentLiabilities/>
                        <r:ownersEquity/>
                        <r:taxDepreciation systemAdjustedValue="" userAdjustedValue=""/>
```

```
                    <r:unTaxedRealisedGainsAndReceipts/>
                    <r:additionsToFixedAssets/>
                    <r:disposalOfFixedAssets/>
                    <r:depreciationRecovered systemAdjustedValue=""
            userAdjustedValue=""/>
                    <r:losses />
                    <r:privateUse systemAdjustedValue="" userAdjustedValue=""/>
                    <r:dividendsPaid/>
                    <r:drawings/>
                    <r:currentAccountYearEndBalances/>
                    <r:taxDeductibleLossOnDisposalOfFixedAssets/>
                    <r:otherAdjustments>
                        <r:adjustments>
                            <r:amount/>
                            <r:description/>
                        <r:adjustments/>
                     <r:otherAdjustments/>
                    <r:yearToDateProvTaxLiability/>
                    <r:thisInstalment/>
                    <r:shareholderProvTax/>
                    <r:refundAmount/>
                    <r:refundIndicator/>
                    <r:creditRecipientList>
                        <r:creditRecipient>
                            <r:recipientIrdNumber/>
                            <r:indicatorType/>
                        <r: creditRecipient />
                     <r: creditRecipientList />
                </rc:formFields>
        </rc:fileBody>
</r:fileRequest>
```

Attribute	Description
systemAdjustedValue	If the system has adjusted the value, the adjustment value should be placed here. The old value will be placed between the field tags.
userAdjustedValue	If the customer has overridden the system calculated adjustment, the adjustment value entered by the customer should be placed here
shareholderProvTax	This field should be left blank. It is intended that this field will enable companies to communicate the amount of provisional tax (if any) they have paid on behalf of shareholders based on proposed future enhancements to the AIM method.
midYearEntry	If a customer is looking to enter AIM mid-year, this must be set to true for the **first filing** of the year only. If the customer is looking to amend their mid-year submission, this must be true.
overFiveMillion	If a customer has an annual income over $5 million this field must be set to true on their first filing of the year
refundAmount	This value is how much has been requested to be refunded. This amount should be set as 0 unless refundIndicator is set to true. If refundIndicator is set to true and refundAmount is set to 0, the entire amount in the account will be refunded.

Attribute	Description
refundIndicator	This field indicates if a balance on the account should be refunded
creditRecipientList	If a customer is paying provisional tax on behalf of other parties, they can provide a list of IRD numbers for their recipients. This will manipulate an indictor on the recipients' account to stop billing notifications using the specified **indicatorType**.
indicatorType	[A, D or E]. This denotes whether the indicator should be [A] Added or [D] Deleted from the recipients' account. An [E] denotes that the indicator currently exists, and no change is required.

The otherAdjustment field is used for any adjustment with no specific field defined. There is a minimum of zero additional adjustments and a maximum of 20 additional adjustments.

For credit transfers, use creditTransferRequest fields in the standardFields portion of the payload.

If the refundIndicator is false and there is no credit transfer request, then the default operation is to hold the entire amount.

At the end of this document there are a number of sample scenarios that illustrate how to manage overpayments.

Inland Revenue
Te Tari Taake

3.2 Prepop

The Prepop operation will be used to acquire a specific subset of fields for a given return. This operation uses the <retrieveFormInfoRequest> structure for the request which will have a unique response across tax types.

<**retrieveFormInfoRequest**> structure:

```
<rc:retrieveFormInfoRequest xmlns:cmn="urn:www.ird.govt.nz/GWS:types/Common.v2"
                xmlns:rc="urn:www.ird.govt.nz/GWS:types/ReturnCommon.v2"
                xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
                xsi:schemaLocation="urn:www.ird.govt.nz/GWS:types/ReturnCommon.v2">
        <cmn:softwareProviderData>
                <cmn:softwareProvider>SoftwareProvider</cmn:softwareProvider>
                <cmn:softwarePlatform>SoftwarePlatform</cmn:softwarePlatform>
                <cmn:softwareRelease>V2</cmn:softwareRelease>
        </cmn:softwareProviderData>
        <cmn:identifier IdentifierValueType="ACCIRD">123456789</cmn:identifier>
        <cmn:accountType>INC</cmn:accountType>
        <rc:periodEndDate>2018-04-30</rc:periodEndDate>
        <rc:majorFormType>SOA</rc:majorFormType>
        <rc:minorFormType>SOA_2</rc:minorFormType>
        <rc:midYearEntry>true</rc:midYearEntry>
</rc:retrieveFormInfoRequest>
```

NOTE: The midYearEntry field is an optional element which only needs to be declared when the user is intending to enter AIM mid-year.

When using the pre-population service for AIM, the tax type will be INC, the majorFormType will be SOA (for Statement of Account) and minorFormType of SOA_2. The response body will only be populated if the customer is eligible for AIM.

<**prepopResponse**> structure:

```
<prepopResponse xmlns="urn:www.ird.govt.nz/GWS:types/ReturnCommon.v2">
        <statusMessage xmlns="urn:www.ird.govt.nz/GWS:types/Common.v2">
                <statusCode>0</statusCode>
                <errorMessage/>
        </statusMessage>
        <responseBody xmlns:r="urn:www.ird.govt.nz/GWS:types/ReturnAIM.v2"
                        xsi:type="r:PrepopResponseBodyType">
                <r:irdNumber>123456789</r:irdNumber>
                <r:filingFrequency>Two Monthly Odd</r:filingFrequency>
                <r:returnPeriodDate>2018-04-30</r:returnPeriodDate>
                <r:returnType>IR4</r:returnType>
                <r:balanceDate>2018-03-31</r:balanceDate>
                <r:periodBalance>1000.54</r:periodBalance>
                <r:totalPenalties>50.63</r:periodBalance>
                <r:totalInterest>5.89</r:periodBalance>
                <r:residualIncomeTax>222</r:residualIncomeTax>
                <r:totalLossCarriedForward>555</r:totalLossCarriedForward>
        </responseBody>
</prepopResponse>
```

Field	Requirement	Description
irdNumber	Required	The IRD number for the customer
filingFrequency	Required	The filing frequency of the statement of activity
returnPeriodDate	Required	The period of the statement
returnType	Required	The income tax return type
balanceDate	Required	The income tax balance date
periodBalance	Required	The income tax period balance
totalPenalties	Required	The penalties owing on the income tax period
totalInterest	Required	The interest owing on the income tax period
residualIncomeTax	Required	The income tax residual income tax
totalLossCarriedForward	Required	The income tax total losses carried forward

Inland Revenue
Te Tari Taake

3.3 RetrieveStatus

The RetrieveStatus operation will allow the status of a given return to be queried. The request and response structures are the same for all tax types.

<**retrieveFormInfoRequest**> structure:

```
<rc:retrieveFormInfoRequest xmlns:cmn="urn:www.ird.govt.nz/GWS:types/Common.v2"
                xmlns:rc="urn:www.ird.govt.nz/GWS:types/ReturnCommon.v2"
                xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
                xsi:schemaLocation="urn:www.ird.govt.nz/GWS:types/ReturnAIM.v2">
        <cmn:softwareProviderData>
                <cmn:softwareProvider>SoftwareProvider</cmn:softwareProvider>
                <cmn:softwarePlatform>SoftwarePlatform</cmn:softwarePlatform>
                <cmn:softwareRelease>V2</cmn:softwareRelease>
        </cmn:softwareProviderData>
        <cmn:identifier IdentifierValueType="ACCIRD">123456789</cmn:identifier>
        <cmn:accountType>INC</cmn:accountType>
        <rc:periodEndDate>2019-04-30</rc:periodEndDate>
        <rc:majorFormType>SOA</rc:majorFormType>
        <rc:minorFormType>SOA_2</rc:minorFormType>
</rc:retrieveFormInfoRequest>
```

<**retrieveStatusResponse**> structure:

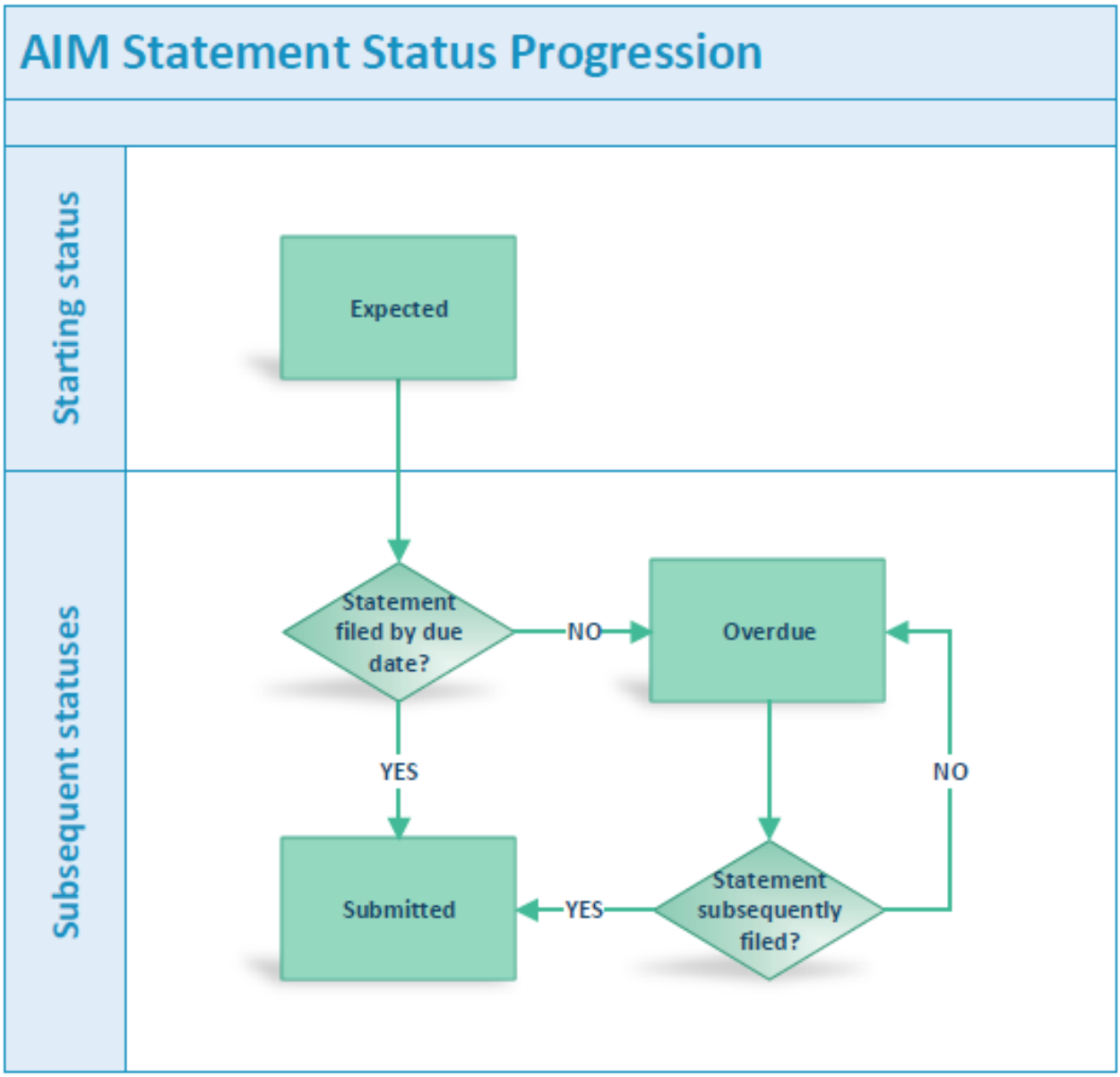
```
<retrieveStatusResponse xmlns="urn:www.ird.govt.nz/GWS:types/ReturnCommon.v2">
        <statusMessage xmlns="urn:www.ird.govt.nz/GWS:types/Common.v2">
                <statusCode>0</statusCode>
                <errorMessage/>
        </statusMessage>
        <responseBody>
                <status code="SUB">Submitted</status>
        </responseBody>
</statusResponse>
```

Although submitted via the Return service, an AIM Statement is not a return and therefore only a subset of the available statuses apply for an AIM Statement.

The following three statuses apply for checking RetrieveStatus for AIM:

Status	Description
Expected	This status is displayed when the filing period has a generated return expectation
Overdue	This status is displayed when the Statement of Activity is overdue
Submitted	This status is displayed when the Statement of Activity is submitted by the customer.

This diagram outlines the process flow of the statuses listed in the table on the previous page.



Inland Revenue
Te Tari Taake

3.4 RetrieveReturn

The retrieveReturn operation allows for any previously-submitted return to be viewed.

<**retrieveFormInfoRequest**> structure:

```
<rc:retrieveFormInfoRequest xmlns:cmn="urn:www.ird.govt.nz/GWS:types/Common.v2"
                xmlns:rc="urn:www.ird.govt.nz/GWS:types/ReturnCommon.v2"
                xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
                xsi:schemaLocation="urn:www.ird.govt.nz/GWS:types/ReturnAIM.v2">
        <cmn:softwareProviderData>
                <cmn:softwareProvider>SoftwareProvider</cmn:softwareProvider>
                <cmn:softwarePlatform>SoftwarePlatform</cmn:softwarePlatform>
                <cmn:softwareRelease>V2</cmn:softwareRelease>
        </cmn:softwareProviderData>
        <cmn:identifier IdentifierValueType="ACCIRD">123456789</cmn:identifier>
        <cmn:accountType>INC</cmn:accountType>
        <rc:periodEndDate>2019-04-30</rc:periodEndDate>
        <rc:majorFormType>SOA</rc:majorFormType>
        <rc:minorFormType>SOA_2</rc:minorFormType>
</rc:retrieveFormInfoRequest>
```

<**retrieveReturnResponse**> structure:

```
<retrieveReturnResponse xmlns="urn:www.ird.govt.nz/GWS:types/ReturnCommon.v2">
        <statusMessage xmlns="urn:www.ird.govt.nz/GWS:types/Common.v2">
                <statusCode>0</statusCode>
                <errorMessage/>
        </statusMessage>
        <responseBody xmlns:r="urn:www.ird.govt.nz/GWS:types/ReturnAIM.v2"
                xsi:type="r:RetrieveReturnResponseBodyType">
           <r:standardFields>
                <isNilReturn xmlns="urn:www.ird.govt.nz/GWS:types/ReturnCommon.v2"/>
           </r:standardFields>
           <r:formFields>
                <r:midYearEntry></r:midYearEntry>
                <r:overFiveMillion></r:overFiveMillion>
                <r:aimInstalmentDate></r:aimInstalmentDate>
                <r:grossSalesAndServiceIncome></r:grossSalesAndServiceIncome>
                <r:openingStock></r:openingStock>
                <r:purchases></r:purchases>
                <r:closingStock systemAdjustedValue="" userAdjustedValue=""></r:closingStock>
                <r:grossProfit></r:grossProfit>
                <r:interestReceived></r:interestReceived>
                <r:dividendsReceived></r:dividendsReceived>
                <r:rentLeaseLicenceIncome></r:rentLeaseLicenceIncome>
                <r:otherIncome></r:otherIncome>
                <r:badDebts></r:badDebts>
                <r:depreciationAndAmortisation></r:depreciationAndAmortisation>
                <r:insurance></r:insurance>
                <r:interestExpense></r:interestExpense>
                <r:fees></r:fees>
                <r:rates></r:rates>
                <r:rentsLeasesLicences></r:rentsLeasesLicences>
                <r:repairsAndMaintenance></r:repairsAndMaintenance>
                <r:researchAndDevelopment></r:researchAndDevelopment>
                <r:relatedPartyRemuneration></r:relatedPartyRemuneration>
                <r:salariesAndWages></r:salariesAndWages>
                <r:contractorPayments></r:contractorPayments>
```

```
                        <r:otherExpenses></r:otherExpenses>
                        <r:exceptionalItems></r:exceptionalItems>
                        <r:netProfitLossBeforeTax></r:netProfitLossBeforeTax>
                        <r:taxAdjustments></r:taxAdjustments>
                        <r:currentYearTaxableProfitLoss></r:currentYearTaxableProfitLoss>
                        <r:accountsReceivable systemAdjustedValue="" userAdjustedValue=""/>
                        <r:cashAndDeposits></r:cashAndDeposits>
                        <r:otherCurrentAssets></r:otherCurrentAssets>
                        <r:vehicles></r:vehicles>
                        <r:plantAndMachinery></r:plantAndMachinery>
                        <r:furnitureAndFittings></r:furnitureAndFittings>
                        <r:land></r:land>
                        <r:buildings></r:buildings>
                        <r:otherFixedAssets></r:otherFixedAssets>
                        <r:intangibles></r:intangibles>
                        <r:sharesAndOwnershipInterests></r:sharesAndOwnershipInterests>
                        <r:termDeposits></r:termDeposits>
                        <r:otherNonCurrent></r:otherNonCurrent>
                        <r:provisions systemAdjustedValue="" userAdjustedValue=""></r:provisions>
                        <r:provisionsForShareholderSalaries adjustedBy=" unadjustedValue=""/>
                        <r:accountsPayable systemAdjustedValue="" userAdjustedValue=""/>
                        <r:currentLoans></r:currentLoans>
                        <r:otherCurrentLiabilities></r:otherCurrentLiabilities>
                        <r:nonCurrentLiabilities></r:nonCurrentLiabilities>
                        <r:ownersEquity></r:ownersEquity>
                        <r:taxDepreciation systemAdjustedValue="" userAdjustedValue=""/>
                        <r:unTaxedRealisedGainsAndReceipts/>
                        <r:additionsToFixedAssets></r:additionsToFixedAssets>
                        <r:disposalOfFixedAssets></r:disposalOfFixedAssets>
                        <r:depreciationRecovered systemAdjustedValue="" userAdjustedValue=""/>
                        <r:losses></r:losses>
                        <r:privateUse systemAdjustedValue="" userAdjustedValue=""></r:privateUse>
                        <r:dividendsPaid></r:dividendsPaid>
                        <r:drawings></r:drawings>
                        <r:currentAccountYearEndBalances/>
                        <r:taxDeductibleLossOnDisposalOfFixedAssets/>
                        <r:otherAdjustments>
                            <r:adjustments>
                               <r:amount/>
                               <r:description/>
                            <r:adjustments/>
                         <r:otherAdjustments/>
                        <r:yearToDateProvTaxLiability></r:yearToDateProvTaxLiability>
                        <r:thisInstalment></r:thisInstalment>
                        <r:shareholderProvTax></r:shareholderProvTax>
                        <r:refundAmount></r:refundAmount>
                        <r:refundIndicator></r:refundIndicator>
                        <r:creditRecipientList>
                            <r:creditRecipientList>
                               <r:recipientIrdNumber/>
                               <r:indicatorType/>
                            <r:creditRecipientList/>
                         <r:creditRecipientList/>
                    </r:formFields>
            </r:responseBody>
</r:retrieveReturnResponse>
```

Inland Revenue
Te Tari Taake

3.5 RetrieveFilingObligation

The retrieveFilingObligation operation is used to retrieve the date on which the next return is due, as well as any overdue returns for a specified account. This operation has the same request and response structure for all tax types.

<**retrieveFilingObligationsRequest**> structure:

```
<rc:retrieveFilingObligationsRequest
                        xmlns:cmn="urn:www.ird.govt.nz/GWS:types/Common.v2"
                        xmlns:rc="urn:www.ird.govt.nz/GWS:types/ReturnCommon.v2"
                        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
                        xsi:schemaLocation="urn:www.ird.govt.nz/GWS:types/ReturnAIM.v2">
        <cmn:softwareProviderData>
                <cmn:softwareProvider>SoftwareProvider</cmn:softwareProvider>
                <cmn:softwarePlatform>SoftwarePlatform</cmn:softwarePlatform>
                <cmn:softwareRelease>V2</cmn:softwareRelease>
        </cmn:softwareProviderData>
        <cmn:identifier IdentifierValueType="ACCIRD">123456789</cmn:identifier>
        <cmn:accountType>INC</cmn:accountType>
        <rc:majorFormType>SOA</rc:majorFormType>
        <rc:minorFormType>SOA_2</rc:minorFormType>
</rc:retrieveFilingObligationsRequest>
```

<**retrieveFilingObligationsResponse**> structure:

```
<retrieveFilingObligationsResponse
xmlns="urn:www.ird.govt.nz/GWS:types/ReturnCommon.v2">
        <statusMessage xmlns="urn:www.ird.govt.nz/GWS:types/Common.v2">
                <statusCode>0</statusCode>
                <errorMessage/>
        </statusMessage>
        <responseBody>
            <filingObligation>
                <periodEndDate>2018-05-31</periodEndDate>
                <status code="OVERDU">Overdue</status>
                <dueDate>2018-06-28</dueDate>
            </filingObligation>
            <filingObligation>
                <periodEndDate>2018-06-30</periodEndDate>
                <status code="EXP">Expected</status>
                <dueDate>2018-07-30</dueDate>
            </filingObligation>
        </responseBody>
</retrieveFilingObligationsResponse>
```

Inland Revenue
Te Tari Taake

4 Additional development resources

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 Schemas

The AIM.v2 schema for the Return Service imports a Common.v2.xsd which has some data types specific to Inland Revenue. This Common.v2.xsd will be used in other Gateway Services outside of the /Returns/ namespace so it must be kept up-to-date, without numerous redundant versions remaining.

The ReturnCommon.v2.xsd imports the Common.v2.xsd and creates data types to be used across all tax types and return types. ReturnCommon.v2.xsd also includes two request elements and two response elements. These requests are retrieveFormInfoRequest and retrieveFilingObligationsRequest, while the responses are retrieveFilingObligationsResponse and retrieveStatusResponse.

The reason for adding root-level elements in the ReturnCommon.v2.xsd is due to the fact that these request and response structures will never change, regardless of the tax type. This allows Inland Revenue to keep a uniform request and response structure across all current and future tax types.

Importing from ReturnCommon.v2.xsd will be schemas that require more fine-grained detail. These will primarily define the request for the File operation, the response for RetrieveReturn and the response for Prepop.

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

4.2 WSDLs

The Returns Gateway Service has one WSDL, which has a target namespace of **https://services.ird.govt.nz/GWS/Returns** and can be found at https://services.ird.govt.nz:4046/gateway/GWS/Returns/?singleWsdl

All WSDL messages follow this naming convention:

```
Return_<operation>_InputMessage or Return_<operation>_OutputMessage.

<wsdl:portType name="Return">
        <wsdl:operation name="File">
        <wsdl:operation name="Prepop">
        <wsdl:operation name="RetrieveStatus">
        <wsdl:operation name="RetrieveFilingObligations">
        <wsdl:operation name="RetrieveReturn">
<wsdl:service name="Return">
```

A development version of the WSDL is provided with this build pack. For easier WSDL consumption, the <xs:any> structure has been replaced with a reference to the corresponding element in the ReturnAIM.v2.xsd. This will allow any tools that consume the WSDL to automatically pull in the data structures from the XSD. To use this, ensure the WSDL provided by Inland Revenue is in the same directory as Common.v2.xsd, ReturnCommon.v2.xsd and ReturnAIM.v2.xsd.

See beginning of section 4 for instructions on where to find WSDLs for this service.

Inland Revenue
Te Tari Taake

5 Responses

The response message from the Gateway Services will always include a status code and status message. These values will describe the successes or failures of your web service call. Following the status message will be the responseBody, which will return the data for the given operation.

5.1 Generic gateway response codes

The following response codes are common to all Gateway Service calls. The operations on the Return Service all have framework security validation applied at Account level and that is reflected in the descriptions of the codes below:

Standard codes	Standard message	Description
-1	An unknown error has occurred	This error will be logged by the Gateway Services and evaluated the next business day
0		0 indicates a successful web service call. Note: 0 does not display a standard message.
1	Authentication failure	Authentication failure means the token provided is not a valid token
2	Missing authentication token(s)	No OAuth token in HTTP header
3	Unauthorised access	The logon making the call does not have access to make the request on behalf of the client or agency
4	Unauthorised delegation	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: • The submitted cmn:identifier has an invalid value. • The identifier type (IdentifierValueType attribute on cmn:identifier) supplied is invalid. • The AccountType supplied does not exist for that identifier. • All the values above are valid, but the provided OAuth token does not have delegated access to that Customer or Account.
5	Unauthorised vendor	The vendor provided is not authorised to use these suite of services

Standard codes	Standard message	Description
7	Account type not supported	This code will be returned for queries on account types not supported in any gateway services web services. For April 2018 this will be any account type other than AIL, AIP, BPA, MPO, CRS, DWT, FAT, FBT, GMD, GSD, GST, INC, IIT, ITN, IPS, NRT, PIE, PRS, PSO, EMP, RLT, RWT. For specific services some of the account types above may not be supported—please see the related documentation and the service-specific response codes below.
20	Unrecognised XML request	The XML submitted is not recognisable and no schema can be determined
21	XML request failed validation	The XML structure did not meet the definition laid out by the schemas published by Inland Revenue
22	Invalid Payload IRD	The external requester submitted an invalid IRD number in the payload body.
(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	An unexpected technical fault has been detected. Depending on the context (eg if an online user is waiting), try the request again after at least five seconds. If the fault recurs then please contact GatewayServices@ird.govt.nz.

Inland Revenue
Te Tari Taake

5.2 Generic returns response codes

The following response codes are specific to Returns Gateway Service calls:

Standard codes	Standard message	Description
100	Invalid request data	Could not extract data from XML payload
101	Unable to file return	An error has occurred while filing return. This may be due to invalid information in the specific return form fields.
102	ID/Account type not valid	The account type/ID submitted does not exist
103	No return found	No return exists on the selected filing period
104	Invalid filing period	Error may be returned for one of the following reasons: • The periodEndDate did not match a valid filing period for the account • Attempting to file a SOA for a filing period that is before the filing period of the most recently filed SOA (SOAs must be filed sequentially, in correct order).
105	No filing obligations found	No valid filing obligations were found. This could be completely acceptable if they were not expecting to have any filing obligations.
106	Operation not available for major form type	The operation performed does not exist for the major form type submitted
140	Invalid Minor Form Type	The minor form type provided is invalid or the minor form type is invalid for the account type.

Inland Revenue
Te Tari Taake

5.3 AIM-specific response codes

The following response codes are specific to AIM Gateway Service calls:

Standard codes	Standard message	Description
110	Customer not identified	The ID submitted does not exist
111	Customer is ineligible for AIM Statement of Activity	Inland Revenue has indicated this Customer in ineligible for AIM
112	Invalid entity type	Valid Customer Subtypes are: "COMPNY", "INDVDL", "SOCITY", "UNTTST" and Customer must not be part of a Consolidated Group
113	Period not provided	Filing Period does not exist or was not provided
114	Invalid period	Period occurs before Gateway Services go-live
115	Instalment date not provided	Instalment date does not exist or was not provided
116	Invalid instalment date	Invalid Instalment date based on provided Filing Period
117	Missing previous statement	Previous Statement must be provided before subsequent Statement (example: March must be filed before April)
118	Duplicate statement of activity	Statement of Activity already exists for provided Customer and Period
119	Customer not enrolled in AIM	Customer tried to retrieve a return when they are not enrolled in AIM
120	Ratio return has been filed for this tax year	Customer has filed a GST Ratio Return this tax year
122	Transitional year processing error	Customer is in an income tax transitional year
123	Income tax account inactive	Customer currently has an inactive income tax account
124	No return has been filed for first period of tax year.	The customer has missed the first SOA filing and will need to correct this before filing this one.
125	Cannot enter AIM mid-year while on Estimate provisional method	Customer must be Ratio, Standard or no provisional method to opt into AIM part way through the year
126	Customer must have a tax transaction balance of 0	The customer must have met all of their provisional obligations up to the run-date in order to switch into AIM
127	Cannot opt into AIM mid-year if a previous Statement of Activity has been filed for the year	To opt in mid-year, it must be the first submission that the customer sends

Standard codes	Standard message	Description
128	Can only amend the most recently filed Statement of Activity	You can only amend the most recently filed Statement of Activity as it is made to declare income as the year progresses
129	Mid-year entry missing from amendment submission.	Customer must keep the mid-year entry Boolean as true if they are amending their initial mid-year filing for AIM
130	Incorrect version of SOA submitted for amendment.	A customer cannot file a SOA using version 1 and then amend using version 2 as they are not backwards compatible

Inland Revenue
Te Tari Taake

6 Example scenarios

6.1 Managing overpayments

Scenario	Example		Statement of Activity
Customer wants the overpaid provisional tax *refunded* in full.	Overpaid provisional tax Refund	$2,000 $2,000	• refundIndicator = true • refundAmount = 0 or 2000 (if refundIndicator is true, and refundAmount is 0, the entire amount will be refunded) • No transfer instructions
Customer wants a portion of the overpaid provisional tax *refunded* and the balance *held* in their income tax account.	Overpaid provisional tax Refund Hold in income tax account	$2,000 $1,500 $ 500	• refundIndicator = true • refundAmount = 1500 • No transfer instructions
Customer wants a portion of the overpaid provisional tax *refunded* and the balance *transferred* to another account(s).	Overpaid provisional tax Refund Transfer to another account	$2,000 $1,200 $ 800	• refundIndicator = true • refundAmount = 1200 • Transfer instructions completed (eg instructions to transfer $800)
Customer wants a portion of the overpaid provisional tax *refunded*, a portion *transferred* to another account(s) and the balance *held* in their income tax account.	Overpaid provisional tax Refund Transfer to another account Hold in income tax account	$2,000 $1,000 $ 200 $ 800	• refundIndicator = true • refundAmount = 1000 • Transfer instructions completed (eg instructions to transfer $200)
Customer wants all of the overpaid provisional tax *held* in their income tax account.	Overpaid provisional tax Hold in income tax account	$2,000 $2,000	• refundIndicator = false • refundAmount = 0 • No transfer instructions
Customer wants a portion of the overpaid provisional tax *held* in their income tax account and a portion *transferred* to another account(s).	Overpaid provisional tax Transfer to another account Hold in income tax account	$2,000 $ 500 $1,500	• refundIndicator = false • refundAmount = 0 • Transfer instructions completed (eg instructions to transfer $500)

Scenario	Example	Statement of Activity
Customer wants all of the overpaid provisional tax *transferred* to another account(s).	Overpaid provisional tax $2,000 Transfer to another account $2,000	• refundIndicator = false • refundAmount = 0 • Transfer instructions completed (eg instructions to transfer $2,000)

Inland Revenue
Te Tari Taake

6.2 'Year to date' provisional liability and 'This instalment'

Example 1:

Customer's accounting income (and therefore their provisional tax liability) is increasing during the year.

The AIM-capable software calculates the year to date provisional tax liability and the amount due (if any) at each instalment date.

Statement of Activity	Instalment 1	Instalment 2	Instalment 3	Instalment 4	Instalment 5	Instalment 6
Year to date provisional tax liability	$1,000	$1,700	$3,000	$10,000	$16,000	$23,000
This instalment	$1,000	$700	$1,300	$7,000	$6,000	$7,000

The year to date provisional tax liability from the Statement of Activity will be recorded in the customer's income tax account along with payments made.

Customer income tax account	Instalment 1	Instalment 2	Instalment 3	Instalment 4	Instalment 5	Instalment 6
Provisional tax liability	$1,000	$1,700	$3,000	$10,000	$16,000	$23,000
Reversal		$1,000-	$1,700-	$3,000-	$10,000-	$16,000-
Payments	$1,000-	$1,000- $700-	$1,000- $700- $1,300-	$1,000- $700- $1,300- $7,000-	$1,000- $700- $1,300- $7,000- $6,000-	$1,000- $700- $1,300- $7,000- $6,000- $7,000-
Refunds						
Balance	$0	$0	$0	$0	$0	$0

The instalment amount from the Statement of Activity will be recorded as the amount due (if any) to build up a record of all instalments for the year.

Customer income tax due dates						
Instalment 1	$1,000	$1,000	$1,000	$1,000	$1,000	$1,000
Instalment 2		$700	$700	$700	$700	$700
Instalment 3			$1,300	$1,300	$1,300	$1,300
Instalment 4				$7,000	$7,000	$7,000
Instalment 5					$6,000	$6,000
Instalment 6						$7,000

Example 2:

Customer's accounting income (and therefore their provisional tax liability) fluctuates during the year.

Statement of Activity	Instalment 1	Instalment 2	Instalment 3	Instalment 4	Instalment 5	Instalment 6
Year to date provisional tax liability	$1,000	$1,700	$3,000	$1,100	$800	$1,500
This instalment	$1,000	$700	$1,300	$0	$0	$700

The year to date provisional tax liability from the Statement of Activity will be recorded in the customer's income tax account along with payments made. Where the payments made up to an instalment date exceed the year to date provisional tax liability, the overpayment will be refunded unless directions have been provided on the Statement of Activity to hold or transfer the overpayment.

Inland Revenue
Te Tari Taake

Customer income tax account	Instalment 1	Instalment 2	Instalment 3	Instalment 4	Instalment 5	Instalment 6
Provisional tax liability	$1,000	$1,700	$3,000	$1,100	$800	$1,500
Reversal		$1,000-	$1,700-	$3,000-	$1,100-	$800-
Payments	$1,000-	$1,000- $700-	$1,000- $700- $1,300-	$1,000- $700- $1,300-	$1,000- $700- $1,300-	$1,000- $700- $1,300- $700-
Refunds				$1,900	$1,900 $300	$1,900 $300
Balance	$0	$0	$0	$0	$0	$0

The instalment amount from the Statement of Activity will be recorded as the amount due (if any) to build up a record of all of the instalments for the year. Previous due dates may be adjusted to ensure the total of all due dates match the year to date provisional tax liability.

Customer income tax due dates						
Instalment 1	$1,000	$1,000	$1,000	$1,000	$800	$800
Instalment 2		$700	$700	$100	$0	$0
Instalment 3			$1,300	$0	$0	$0
Instalment 4				$0	$0	$0
Instalment 5					$0	$0
Instalment 6						$700

Inland Revenue
Te Tari Taake

6.3 Ledger, software-generated and user-entered values

Example 1:

Ledger entry	Software generated adjustment	User entered adjustment
Balance (if any) in the ledger for an item.	Adjustment that needs to be made to a ledger amount, calculated by AIM-capable software based on the relevant Determination.	Amount (if any) entered by the customer as a more appropriate adjustment based on individual circumstances.
Example: ***Provisions balance in the ledger is $2,000.*** **Provisions = 2000**	*Software calculated adjustment for provisions is $500 to take it to a AIM suitable amount of $2,500.* systemAdjustedValue = 500	*Customer enters adjustment of $450 as they consider the amount should be $2,450* userAdjustedValue = 450
Example: ***Provisions balance in the ledger is $2,000.*** **Provisions = 2000**	*Software calculated adjustment for provisions is nil.* systemAdjustedValue = 0	*Customer enters adjustment of $450.* userAdjustedValue = 450
Example: ***Provisions balance in the ledger is $2,000.*** **Provisions = 2000**	*Software has insufficient information to calculate an adjustment for provisions.* systemAdjustedValue = blank	*Customer enters adjustment of $450.* userAdjustedValue = 450
Example: ***There is no balance for provisions in the ledger.*** **Provisions = 0**	*No adjustment for provisions is calculated.* systemAdjustedValue = blank	N/A userAdjustedValue = blank

Inland Revenue
Te Tari Taake

7 Change log

This table lists all changes that have been made to this build pack (most recent changes listed first).

Date of change	Document section	Description
29/06/21	2.4	Security section restructured – now contains sub-sections on information classification, transport layer security and certificates, ciphers, end points, and authentication and authorisation
	2.4.2	New information added to provide for Inland Revenue's support for TLS1.3, and deprecation of certain TLS1.2 ciphers
		Updated end point information on web service consumer identification for desktop connections (in table)
		Updated list of recommended certificate authorities
		Updated list of requirements for accepting public X.509 keys – now includes ECDSA
	1	Moved 'Mutual Transport Layer Security and certificates' section into section 2.4
		'Prerequisites' table removed and absorbed into section 2.4.2
	4	Renamed 'End points, schema and WSDLs' section to 'Additional development resources'
		Removed section with redundant reference to end points
	5.1	Updated description of following response code: (none) (soap fault) UnAuthorised
	7	Glossary removed
08/07/20	3.1	Increased maximum number of transfers from 10 to 20. Also updated ReturnCommon.v2.xsd with this change
19/05/20	1.1	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	Updated instructions on where to find related build packs.
	4	Removed boxed instructions on where to find current end points, schemas and WSDLs and updated with new instructions.
	4.3	Removed redundant note at end of section regarding WSDLs. Added following text: *See beginning of section 4 for instructions on where to find WSDLs for this service.*

Date of change	Document section	Description
	4.1	Text updated to this: *See instructions above for where to find end points for this service.*
	4.2	Note added: *See instructions at beginning of section 4 for where to find schemas for this service.*
	2.4	Note added at end of section: *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	Updated hyperlink in boxed text at start of section
27/01/20	3.1	Updated note in description of isAmended field: **NOTE:** If isAmended=true then amendReason and amendDetails are required. Otherwise empty values are required in the amendReason and amendDetails fields.
		Changed requirement details in amendReason and amendDetails fields from 'optional' to 'conditional'
	2.4	Updated TLS information to reflect use of TLS1.3
	1.4	Note added to Prerequisites table: Note that the same certificate cannot be used for the Test and Production environments.
07/10/19	3	Updated field description for **IdentifierValueType:** The ID type being submitted. This can be ACCIRD, NZBN or ACC. The value submitted for this field should contain only digits, with no dashes (with the exception of the ACC field, which may contain letters to identify the account type). IRD Numbers that are eight digits must be padded with a leading zero.
13/09/19	3 Glossary	Changed ACCID to ACC
05/09/19	2.3 3.2 3.3 3.4	Changed formInfoRequest to retrieveFormInfoRequest
8/04/19	2.4	Sentence removed: *For TDS Real Time web service requests, an OAuth access token is required in the HTTP header.*
	5.1	Error code 6 removed (authentication expired)
	7	API added to glossary

Date of change	Document section	Description
	1.3	• 1.3 section heading changed from 'Related build packs' to 'Related services' • 1.3.1 Identity and Access Services wording changed • 1.3.2 section added to include Intermediation Service
14/03/19	5.2	Description expanded for error code 104 Invalid filing period
11/03/19	3.5 5.2 5.3	Added <rc:**minorFormType**>SOA_2</rc:minorFormType> to the obligation request. Added error codes 140 and 130.
26/02/19	3	Description of softwarePlatform in table removed: *"The software package that is making the request".* Replaced with new description: "The field value will be provided by Inland Revenue during the onboarding process."
25/01/19	3.1 3.2 3.3 3.4	Added to request payload in each section: <rc:**minorFormType**>SOA_2</rc:minorFormType>
20/12/18		**AIM Return Service build pack V2.0 created**