Inland Revenue
Te Tari Taake

Inland Revenue

Build Pack: Return Service—Employment Information

Date: 25/08/2020
Version: 2.0

Inland Revenue
Te Tari Taake

Contents

V2 SERVICE UPDATES 3
Amendment scenarios—Differences between EI v1 and v2 4
1 Overview 5
1.1 This solution 5
1.2 Intended audience 5
1.3 Related services 5
1.3.1 Identity and Access Services (required) 5
1.3.2 Intermediation Service (recommended) 5
1.4 Prerequisites 6
1.4.1 Mutual Transport Layer Security and certificates 6
2 Solution design 7
2.1 Architecture 7
2.2 Service scope 7
2.3 Messaging 7
2.4 Security 9
3 Operations 12
3.1 File 14
3.1.1 Amendment methods 21
3.2 RetrieveStatus 22
3.3 RetrieveReturn 26
3.4 Prepop 28
4 End points, schemas and WSDLs 30
4.1 End points 30
4.2 Schemas 30
4.3 WSDLs 31
5 Responses 32
5.1 Generic Gateway response codes 32
5.2 Generic returns response codes 33
5.3 Employment Information-specific response codes 34
6 Glossary 36
7 Change log 38

Inland Revenue
Te Tari Taake

V2 SERVICE UPDATES

The following key changes have been made to the Employment Information Return Service in preparation for the update to v2.

Document section	Description
3 Operations	• Updated schema to say 'ReturnEI.V2.xsd', Common.v2.xsd, and ReturnCommon.v2.xsd

3.1 File	• Requirement for amendReason and amendDetails changed to 'conditional'. If isAmended=true then amendReason and amendDetails are required. Otherwise empty values are required in the amendReason and amendDetails fields.
	• Updated existing field employeePayFrequency from 'Optional' to 'Required'
	• Added new optional field 'hoursPaid' (EI line item)
	• Added two new optional fields 'priorPeriodGrossAdjustment' and 'priorPeriodPAYEAdjustment' (EI line items)
	• Added two new optional fields 'totalPriorPeriodGrossAdjustment' and 'totalPriorPeriodPAYEAdjustment'
	• Added three new optional fields 'essEarnings', 'slcirDeductions', 'slborDeductions' (EI line items)
	• Updated employeeName to allow 255 characters, previously this was 20
	• Removed values from TaxCode table: ESS,SLCIR,SLBOR
	• Added three new optional fields:
	  ○ totalESSEarnings
	  ○ totalSLCIRDeductions
	  ○ totalSLBORDeductions.
	• Added note to clarify use of childSupportCode
	• There is a behaviour change between EI v1 and EI v2. In EI v1, a return can be filed and then immediately amended after receiving a successful response. An EI v2, a return must be processed in order to be amended. Returns will process within 5 minutes.

3.3 Retrieve Return	• New optional fields above will also be included in the Retrieve Return response • Added note to Retrieve Return to clarify that it will return all existing fields on the return
3.4 Prepop	• Updated employeeName to allow 255 characters, previously this was 20
4.3 Retrieve Status	• Responses now include minorFormType • SubmissionKey is now optional in the request body • Now supports multiple return statuses (repeating elements)

Document section	Description
5.3 Employment Information-specific response codes	• Added new response code 170: 'The provided tax code is invalid'. Used to detect when tax code supplied is either ESS, SLCIR or SLBOR. • Added response code 144: Return being submitted • Removed response code 166: EI temporarily locked for processing, not needed in V2

Amendment scenarios—Differences between EI v1 and v2

Due to the introduction of Employment Information (EI) Gateway Service v2, there are specific rules that service providers must adhere to when submitting amendments for an existing EI return on a period. These rules apply both to EI returns submitted through Gateway Service v2, as well as returns that are submitted through EI Gateway Service v2.

In order to support amending prior EI returns, both EI v1 and EI v2 must be supported by service providers.

The EI version can be identified in the standard header's majorFormType field:

- EI version 1 = "EI"
- EI version 2 = "EI2".

Original Return	Amended Return (via Gateway)	Result	Action
EI v1	EI v1	Success	
	EI v2	Failure – EI v2 Response Code 169: Submitted incorrect EI version (please refer to EI v2 build pack for more information on this response code).	This action is not allowed. Returns submitted through EI v1 must be amended with EI v1.
EI v2	EI v1	Failure – EI v1 Response Code 169: Submitted incorrect EI version (please refer to EI v1 build pack for more information on this response code).	This action is not allowed. Returns submitted through EI v2 must be amended with EI v2.
	EI v2	Success	

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 Employment Information (EI) 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 of 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 EI Return 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 services 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.

This Return Service build pack was written using information from v1.5 of the IAS build pack.

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.

1.4 Prerequisites

Party	Requirement	Description
Service provider	Acquire a X.509 certificate from a competent authority for the Test and Production environments	This is required when using mutual TLS with cloud-based service 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 EI Return 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.

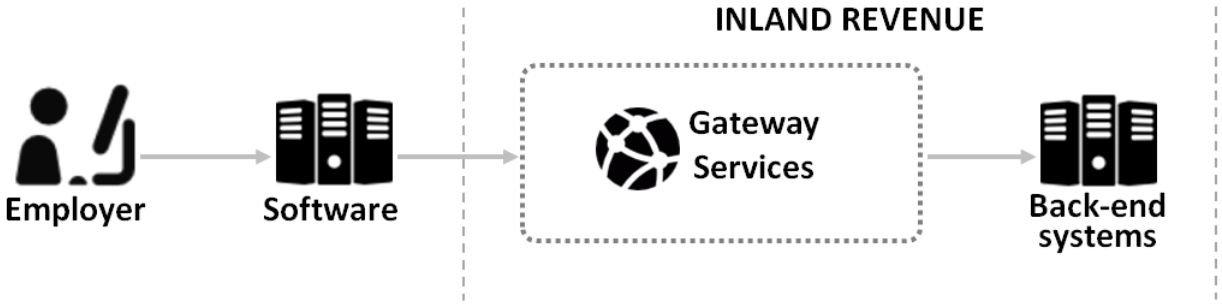
- Comodo
- GeoTrust
- DigiCert
- GlobalSign
- Symantec
- Thawte
- IdenTrust
- Entrust
- Network Solutions
- RapidSSL
- Entrust Datacard
- GoDaddy.

2 Solution design

2.1 Architecture

Inland Revenue is offering a suite of web services to facilitate interactions with Inland Revenue via software packages. The Gateway Services suite will be used by approved software vendors 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 employer to Inland Revenue.



The 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.

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

2.2 Service scope

The EI Return Service supports the following operations:

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

2.3 Messaging

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

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: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:To>https://services.ird.govt.nz/Gateway/GWS/Returns</a:To>
        <a:Action>https://services.ird.govt.nz/GWS/Returns/Return/Operation</a:Action>
    </soap:Header>
    <soap:Body>
        <ret:Prepop>
            <ret:ReturnPrepopRequestMsg>
                <prep:PrepopRequestWrapper>
                    <rc:formInfoRequest xmlns:xsi…
                        <…PrepopFields…>
                    </rc:formInfoRequest>
                </prep:PrepopRequestWrapper>
            </ret:ReturnPrepopRequestMsg>
        </ret:Prepop>
    </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">
    <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></errorMessage>
                    </statusMessage>
                </fileResponse>
            </b:FileResponseWrapper>
        </FileResult>
        </FileResponse>
    </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 IAS build pack.

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 service 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 service providers from issues caused by heavy usage from other providers.

2. For service 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 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 setup 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
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)

	End point for cloud-based connections	End point for desktop connections
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 different client_id/client_secret credentials to cloud application clients
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

Delegated permissions: The services will allow one to retrieve all of the data for a customer that the calling user (as represented by the OAuth token) has access to. There may be additional accounts this identity does not have access to, those will not be mentioned. If an account or data in it is targeted by the request parameters but the user does not have permission an error will be returned. This access will depend on delegation permissions set up in myIR. If the token represents a user in a tax agency or other intermediary, then the agent-client linking is also considered.

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.v2.xsd
- rc: ReturnCommon.v2.xsd
- r: returnSpecific.xsd (for example ReturnEI.v2.xsd)

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, as long as 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 for Inland Revenue front-line staff (such as in the telephone contact centre) to search.

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
- 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>v1</cmn:softwareRelease>
</cmn:softwareProviderData>
<cmn:identifier IdentifierValueType="ACCIRD">012345678</cmn:identifier>
<cmn:accountType>EMP</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 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. Format examples: • IRD: 123456789 • ACC: 123456789EMP001
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 (EMP)

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 all EI 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>Software1</cmn:softwareProvider>
            <cmn:softwarePlatform>Software1Package</cmn:softwarePlatform>
            <cmn:softwareRelease>v1</cmn:softwareRelease>
      </cmn:softwareProviderData>
      <cmn:identifier IdentifierValueType="ACCIRD">012345678</cmn:identifier>
      <cmn:accountType>EMP</cmn:accountType>
      <rc:periodEndDate>2017-03-31</rc:periodEndDate>
      <rc:majorFormType>EI2</rc:majorFormType>
  </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	Last day of the pay date month
majorFormType	Required	The form type (EI2)

Example scenario:

- Attempting to submit an EI return for the 2018-January period.
  <cmn:accountType>EMP</cmn:accountType>
  <rc:periodEndDate>2018-01-31</rc:periodEndDate>
  <rc:majorFormType>EI2</rc:majorFormType>

<**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.

<**StandardFields**> structure:

```
<r:fileRequest namespaces...>
     <rc:fileHeader>...</rc:fileHeader>
     <rc:fileBody>
        <rc:standardFields>
                <rc:isNilReturn>false</rc:isNilReturn>
                <rc:isFinalReturn>false</rc:isFinalReturn>
                <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
isFinalReturn	Optional	This field notes that the account is ready to be closed after this filing period
isAmended	Required	This allows for a return to be filed as an amendment. NOTE: If isAmended=true then amendReason and amendDetails are required.
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 10 can be submitted for every file operation. NOTE: Credit transfer requests are not supported and will result in error code: 150 - Credit transfer requests are not supported

Proper uses:

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

Example scenario:

- Attempting to amend an EI return due to lack of information from client.

```
<rc:isNilReturn>false</rc:isNilReturn>
<rc:isFinalReturn>false</rc:isFinalReturn>
<rc:isAmended>true</rc:isAmended>
<rc:amendReason>KEY</rc:amendReason>
<rc:amendDetails>Client updated pay information</rc:amendDetails>
```

Multiple EI returns can be submitted for each payDayDate.

Note that an employer cannot commence payday filing part way through a month—payday filing must start with the very first payday in the month.

<**FormFields**> structure (for EI):

```
<r:fileRequest xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:cmn="urn:www.ird.govt.nz/GWS:types/Common.v2"
        xmlns:rc="urn:www.ird.govt.nz/GWS:types/ReturnCommon.v2"
        xmlns:r="urn:www.ird.govt.nz/GWS:types/ReturnEI.v2"
        xsi:schemaLocation="urn:www.ird.govt.nz/GWS:types/ReturnEI.v2">
   <rc:fileHeader>…</rc:fileHeader>
   <rc:fileBody>
        <rc:standardFields>…</rc:standardFields>
```

Inland Revenue
Te Tari Taake

```
                <rc:formFields xsi:type="r:FormFieldsType">
                  <r:submissionKey/>
                  <r:isReverseReplace/>
                  <r:payDayDate/>
                  <r:piIrdNumber/>
                  <r:contactName/>
                  <r:contactPhoneNumber/>
                  <r:contactEmail/>
                  <r:employeeFields>
                        <r:employee>
                              <r:lineNumber/>
                              <r:referenceId/>
                              <r:irdNumber/>
                              <r:employeeName/>
                              <r:taxCode/>
                              <r:payPeriodStartDate/>
                              <r:payPeriodEndDate/>
                              <r:employmentStartDate/>
                              <r:employmentFinishDate/>
                              <r:employeePayFrequency/>
                              <r:grossEarnings/>
                              <r:earningsNotLiableACC/>
                              <r:lumpSumIndicator/>
                              <r:payeSchedularTaxDeductions/>
                              <r:childSupportCode/>
                              <r:childSupportDeductions/>
                              <r:studentLoansDeductions/>
                              <r:kiwisaverEmployerContributions/>
                              <r:kiwisaverDeductions/>
                              <r:essEarnings/>
                              <r:slcirDeductions/>
                              <r:slborDeductions/>
                              <r:taxCreditPayrollDonations/>
                              <r:esctDeducted/>
                              <r:familyTaxCredits/>
                              <r:hoursPaid/>
                              <r:priorPeriodGrossAdjustment/>
                              <r:priorPeriodPAYEAdjustment/>
                        </r:employee>
                  </r:employeeFields>
                  <r:totalGrossEarnings/>
                  <r:totalEarningsNotLiableACC/>
                  <r:totalPAYESchedularTaxDeductions/>
                  <r:totalChildSupportDeductions/>
                  <r:totalStudentLoansDeductions/>
                  <r:totalKiwisaverEmployerContributions/>
                  <r:totalKiwisaverDeductions/>
                  <r:totalESSEarnings/>
                  <r:totalSLCIRDeductions/>
                  <r:totalSLBORDeductions/>
                  <r:totalTaxCreditPayrollDonations/>
                  <r:totalESCTDeducted/>
                  <r:totalFamilyTaxCredits/>
                  <r:totalAmountPayable/>
                  <r:totalPriorPeriodGrossAdjustment/>
                  <r:totalPriorPeriodPAYEAdjustment/>
            </rc:formFields>
      </rc:fileBody>
</r:fileRequest>
```

Fields	Requirement	Description
submissionKey	Optional	This field only be used when amending EI returns
isReverseReplace	Optional	This field is required on amendments. Signifies if amendment method is reverse/replace.
payDayDate	Required	Payday means the day on which an employer makes a PAYE income payment to an employee. NOTE: The payday date must be within the same month as the period end date.
piIrdNumber	Optional	PAYE intermediary IRD number
contactName	Optional	Name of payroll contact person
contactPhoneNumber	Optional	Payroll contact phone number
contactEmail	Optional	Payroll contact email
lineNumber	Optional	Inland Revenue-generated unique line item identifier. This field should only be used when amending EI returns.
referenceId	Optional	Unique employee reference field created by the software provider. Can be submitted on initial file, then used as the unique line item identifier for amended lines instead of using the Inland Revenue-generated lineNumber.
irdNumber	Required	IRD number of the employee—it can be zeros if IRD number not known
employeeName	Required	Name of the employee (255 characters)
taxCode	Required	The tax code of the employee (see table below for list of acceptable values)
employmentStartDate	Optional	Employee start date
employmentFinishDate	Optional	Employee end date NOTE: Field not to be supplied if there is no end date
payPeriodStartDate	Required	The first day of the pay period for which the employee was paid. NOTE: The pay period start date cannot be after the pay period

Fields	Requirement	Description
		end date, otherwise a 104 error will be returned.
payPeriodEndDate	Required	The last day of the pay period on which the employee was paid
employeePayFrequency	Required	Pay frequency/cycle for employee (see table below for list of acceptable values)
grossEarnings	Optional	Gross earnings/scheduler payments
earningsNotLiableACC	Optional	Earnings not liable for ACC earner's levy
lumpSumIndicator	Optional	Boolean for indicating lump sum
payeSchedularTaxDeductions	Optional	PAYE/scheduler tax deductions
childSupportCode	Optional	Child support code (see table below for list of acceptable values) NOTE: If there is no value for childSupportCode then omit this element.
childSupportDeductions	Optional	Child support deductions (sequenced with child support code)
studentLoansDeductions	Optional	Student loan deductions
kiwisaverEmployerContributions	Optional	Net KiwiSaver employer contributions
kiwisaverDeductions	Optional	KiwiSaver employee deductions
essEarnings	Optional	Employee share scheme deduction (ESS tax code equivalent)
slcirDeductions	Optional	Student loans compulsory additional deduction (SLCIR tax code equivalent)
slborDeductions	Optional	Student loans voluntary deduction (SLBOR tax code equivalent)
taxCreditPayrollDonations	Optional	Tax credits for payroll donations
esctDeducted	Optional	Employer superannuation contribution tax
familyTaxCredits	Optional—to be filled in by MSD ONLY	Family tax credits
hoursPaid	Optional	Number of hours paid to the employee in the pay period

Fields	Requirement	Description
priorPeriodGrossAdjustment	Optional	Adjustment to the Gross Earnings made in a prior period
priorPeriodPAYEAdjustment	Optional	Adjustment to the PAYE/Schedular Tax Deductions made in a prior period
totalGrossEarnings	Compulsory	Total gross earnings
totalEarningsNotLiableACC	Compulsory	Total earnings not liable for ACC Earner's Levy
totalPAYESchedularTaxDeductions	Compulsory	Total PAYE (including tax on schedular payments)
totalChildSupportDeductions	Compulsory	Total child support deductions
totalStudentLoanDeductions	Compulsory	Total student loan deductions
totalKiwisaverEmployerContributions	Compulsory	Total net KiwiSaver employer contributions
totalKiwisaverDeductions	Compulsory	Total KiwiSaver deductions
totalESSEarnings	Optional	The total of all employees' ESS Earnings
totalSLCIRDeductions	Optional	The total of all employees' SLCIR Deductions
totalSLBORDeductions	Optional	The total of all employees' SLBOR Deductions
totalTaxCreditPayrollDonations	Compulsory	Tax credit payroll donations total
totalESCTDeductions	Compulsory	Total ESCT deducted
totalFamilyTaxCredits	Compulsory	Total Family Tax Credits
totalAmountPayable	Optional	Total amount payable
totalPriorPeriodGrossAdjustment	Optional	Total Gross Earnings adjustments from prior periods
totalPriorPeriodPAYEAdjustment	Optional	Total PAYE/Schedular Tax Deduction adjustments from prior periods

Tax codes		
CAE	**EDW**	**ND**
MESL	**MSL**	**SH**
SB	**SBSL**	**ST**
WT	**SSL**	**ME**
NSW	**M**	**SHSL**
STC	**S**	**STSL**

Child support code	Description
C	Ceased Employment
A	Advanced Payment
P	Protected Earnings
S	Short Term Absence
D	Deducted Previously
O	Other

Employee pay frequency	Description
WK	Weekly
4W	Four-weekly
FT	Fortnightly
MT	Monthly
DA	Daily
AH	Ad hoc/Irregular
HM	Half-monthly (twice a month)

A submissionKey will be provided on file operations of EI returns. This submissionKey will be used to identify the specific EI return on a given filing period for amendment, status and retrieve return requests.

3.1.1 Amendment methods

There are three ways to amend an EI return. All three necessitate populating submissionKey, setting isAmended to true, and filling out amendReason/amendDetails.

1. **lineNumber method:** lineNumbers must be provided for amended lines, which are obtained by the retrieveReturn operation. Only lines that require amendments should be provided. New line items should not include line numbers.

2. **referenceId method:** referenceId must be provided for amended lines, referenceId field are submitted with the initial return. referenceId is then used to uniquely identify individual line items. Only lines that require amendments should be provided. New line items can be submitted using a new referenceId. There is no way to amend the referenceId itself with this method. If a new referenceId is submitted in an attempt to amend a previously submitted referenceId, that line will be treated as new. If both lineNumber and referenceId are submitted on the same line, referenceId will take priority. If lineNumber and referenceId are submitted on separate lines but reference the same line item, then response code 131—'Duplicate line items'—will be returned. The duplicate line item references must be removed before submission is allowed.

3. **Reverse/Replace method:** isReverseReplace must be set to true. The previous return on file for the specified submissionKey will be replaced with this return. Rules work exactly as if it were a new submission. All line items, not just lines that require amendments, must be provided. lineNumbers should not be provided and referenceId is optional.

Sample response on successful EI file operations:

```
<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>
            <submissionKey>1829404672</submissionKey>
        </responseBody>
</fileResponse>
```

3.2 RetrieveStatus

The RetrieveStatus operation will allow the status of a given return to be queried.

<**retrieveEIRequest**> structure:

```
<r:retrieveEIRequest xmlns:cmn="urn:www.ird.govt.nz/GWS:types/Common.v2"
        xmlns:rc="urn:www.ird.govt.nz/GWS:types/ReturnCommon.v2"
        xmlns:r="urn:www.ird.govt.nz/GWS:types/ReturnEI.v2"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="urn:www.ird.govt.nz/GWS:types/ReturnEI.v2">
    <cmn:softwareProviderData>
        <cmn:softwareProvider>SoftwareCompany</cmn:softwareProvider>
        <cmn:softwarePlatform>SoftwarePlatform</cmn:softwarePlatform>
        <cmn:softwareRelease>V1.1</cmn:softwareRelease>
    </cmn:softwareProviderData>
    <cmn:identifier IdentifierValueType="ACCIRD">123456789</cmn:identifier>
    <cmn:accountType xmlns:com="urn:www.ird.govt.nz/GWS:types/Common.v2" >EMP</cmn:accountType>
    <rc:periodEndDate>2018-01-31</rc:periodEndDate>
    <rc:majorFormType>EI2</rc:majorFormType>
        <r:payDayDate>2018-01-12</r:payDayDate>
        <r:submissionKey>12345678</r:submissionKey>
</r:retrieveEIRequest>
```

Field	Requirement	Description
submissionKey	Optional	A unique identifier used to target a specific return

<**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></errorMessage>
        </statusMessage>
        <responseBody>
                <returnStatus>
                  <status code="OPRCD">Ontime-processed</status>
                  <submissionKey>1861250688</submissionKey>
                  <minorFormType>EI2</minorFormType>
                </returnStatus>
                <returnStatus>
                  <status code="OPRCD">Ontime-processed</status>
                  <minorFormType>EI2</minorFormType>
                </returnStatus>
              </responseBody>
        </responseBody>
</statusResponse>
```

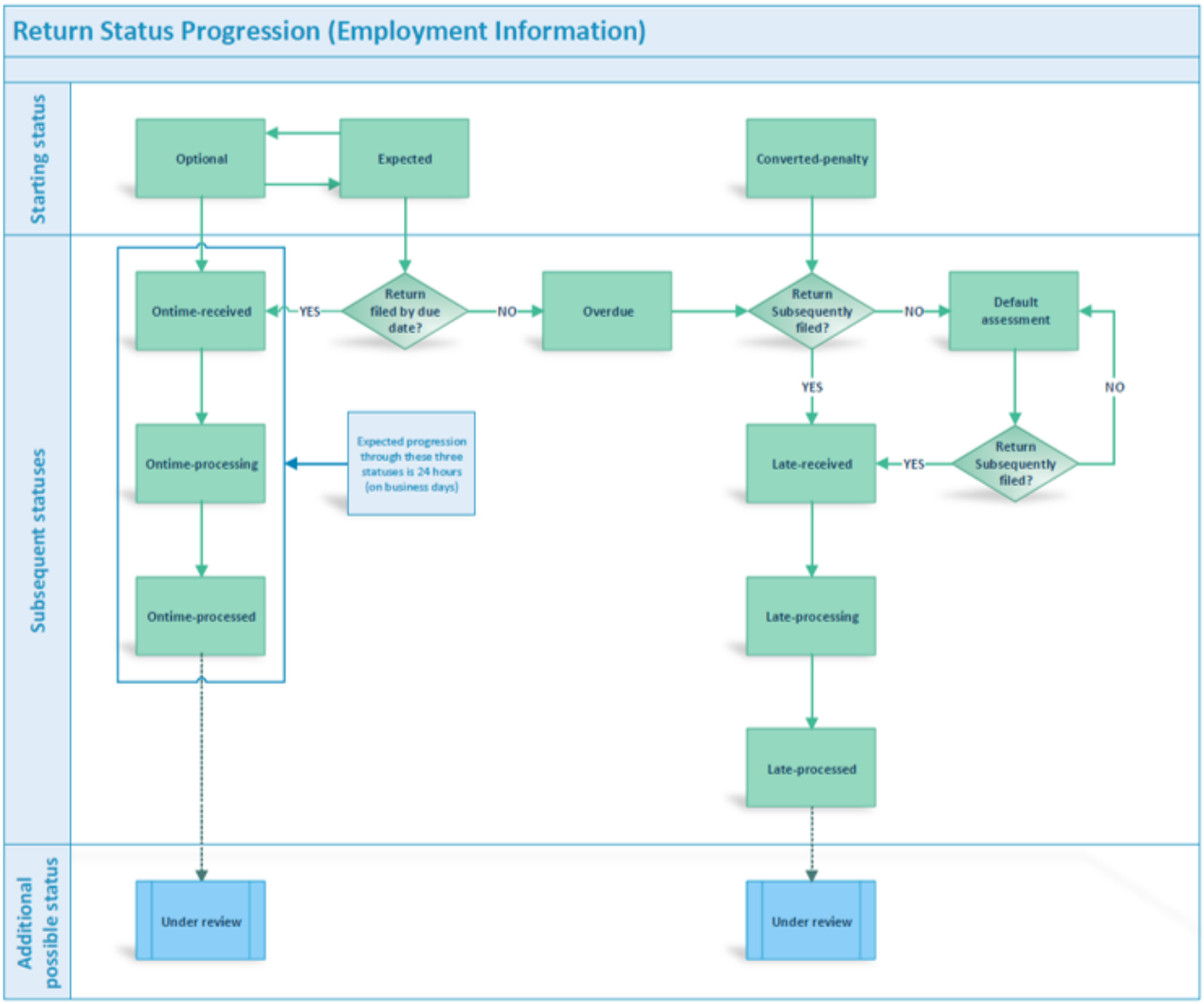
Field	Requirement	Description
status.code	Required	A code representing the status of the return
status	Required	A description of the return's status
submissionKey	Optional	A submissionKey of the return, if available
minorFormType	Required	The majorFormType supplied in the File operation will be returned in the minorFormType of the RetrieveStatus with either: • EI • EI2.

Status	Description
Amended	This status is displayed when the return is amended
Under review	This status is displayed when the return is included in audit
Default assessment	This status is displayed when the return is a default assessment
Converted-penalty	This status is displayed when the return was not filed but created in the conversion process to house a late file penalty
Expected	This status is displayed when the return has generated return expectation
Processing	This status is displayed when the return is processing
Late-processing	This status is displayed when the return is received late and processing
Late-processed	This status is displayed when the return is processed late
Late-received	This status is displayed when the return is received late
New	This status is displayed when the return is not processed

Status	Description
Optional	This status is displayed when the return is not required to be filed, but the customer may choose to file anyway
Ontime-processing	This status is displayed when the return is received on time and processing
Ontime-processed	This status is displayed when the return is processed on time
Ontime-received	This status is displayed when the return is received on time
Overdue	This status is displayed when the return is overdue
Submission received	This status is displayed when the EI return has been received and initial processing has completed. During co-existence, the EI return will be held and forwarded for further processing at the end of the period.
Submitted	This status is displayed when the return is submitted by the customer
Posted	This status is displayed when the return is posted
Processed	This status is displayed when the return is processed

NOTE: Once a return has been filed it can take up to 24 hours for the status to change from 'submitted'.

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



All retrieve operations for EI use <retrieveEIRequest> instead of the standard <retrieveFormInfoRequest>. retrieveEIRequest contains the payDayDate and submissionKey fields that are used to identify a return.

<**retrieveEIRequest**> structure:

```
<r:retrieveEIRequest xmlns:cmn="urn:www.ird.govt.nz/GWS:types/Common.v2"
                     xmlns:rc="urn:www.ird.govt.nz/GWS:types/ReturnCommon.v2"
                     xmlns:r="urn:www.ird.govt.nz/GWS:types/ReturnEI.v2"
                     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
                     xsi:schemaLocation="urn:www.ird.govt.nz/GWS:types/ReturnEI.v2">
          <cmn:softwareProviderData>
                     <cmn:softwareProvider>SoftwareCompany</cmn:softwareProvider>
                     <cmn:softwarePlatform>SoftwarePlatform</cmn:softwarePlatform>
                     <cmn:softwareRelease>V1.1</cmn:softwareRelease>
          </cmn:softwareProviderData>
          <cmn:identifier IdentifierValueType="ACCIRD">123456789</cmn:identifier>
          <cmn:accountType>EMP</cmn:accountType>
          <rc:periodEndDate>2018-01-31</rc:periodEndDate>
          <rc:majorFormType>EI2</rc:majorFormType>
        <r:payDayDate>2018-01-01</r:payDayDate>
        <r:submissionKey>34534523</submissionKey>
</r:retrieveEIRequest>
```

Field	Requirement	Description
payDayDate	Required	The pay day period of the return
submissionKey	Required	The submissionKey is required for the retrieveStatus operation. Only the status of the return with the corresponding submissionKey will displayed.

3.3 RetrieveReturn

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

All retrieve operations for EI use <retrieveEIRequest> instead of the standard <retrieveFormInfoRequest>. retrieveEIRequest contains the payDayDate and submissionKey fields that are used to identify a return.

<**retrieveEIRequest**> structure:

```
<r:retrieveEIRequest xmlns:cmn="urn:www.ird.govt.nz/GWS:types/Common.v2"
                     xmlns:rc="urn:www.ird.govt.nz/GWS:types/ReturnCommon.v2"
                     xmlns:r="urn:www.ird.govt.nz/GWS:types/ReturnEI.v2"
                     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
                     xsi:schemaLocation="urn:www.ird.govt.nz/GWS:types/ReturnEI.v2">
          <cmn:softwareProviderData>
                     <cmn:softwareProvider>SoftwareCompany</cmn:softwareProvider>
                     <cmn:softwarePlatform>SoftwarePlatform</cmn:softwarePlatform>
                     <cmn:softwareRelease>V1.1</cmn:softwareRelease>
          </cmn:softwareProviderData>
          <cmn:identifier IdentifierValueType="ACCIRD">123456789</cmn:identifier>
          <cmn:accountType>EMP</cmn:accountType>
          <rc:periodEndDate>2018-01-31</rc:periodEndDate>
          <rc:majorFormType>EI2</rc:majorFormType>
          <r:payDayDate>2018-01-01</r:payDayDate>
          <r:submissionKey>34534523</r:submissionKey>
</r:retrieveEIRequest>
```

Field	Requirement	Description
payDayDate	Required	The pay day period of the return
submissionKey	Optional	If submissionKey is provided, only the return with that submission key will be displayed. If only a payDayDate is provided, all returns with the corresponding payDayDate will be provided.

<**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></errorMessage>
        </statusMessage>
        <responseBody xmlns:r="urn:www.ird.govt.nz/GWS:types/ReturnEI.v2"
                        xsi:type="r:RetrieveReturnResponseBodyType">
           <r:standardFields>
                 <isNilReturn xmlns="urn:www.ird.govt.nz/GWS:types/ReturnCommon.v2"/>
           </r:standardFields>
           <r:formFields>
                <r:employeeFields>
                        <r:employee>
                                <r:irdNumber></r:irdNumber>
                                <r:employeeName></r:employeeName>
                                <r:taxCode></r:taxCode>
                                <r:payPeriodStartDate></r:payPeriodStartDate>
                                <r:payPeriodEndDate></r:payPeriodEndDate>
                        </r:employee>
                        <r:employee>
                                <r:irdNumber></r:irdNumber>
                                <r:employeeName></r:employeeName>
                                <r:taxCode></r:taxCode>
                                <r:payPeriodStartDate></r:payPeriodStartDate>
                                <r:payPeriodEndDate></r:payPeriodEndDate>
                        </r:employee>
                </r:employeeFields>
            </r:formFields>
        </r:responseBody>
</r:retrieveReturnResponse>
```

NOTE: The response above is an example and contains only the fields required on the return. The retrieveReturnResponse will retrieve all fields on the return, including optional fields if they exist.

3.4 Prepop

The Prepop operation will be used to acquire a specific subset of fields for a given return. All retrieve operations for EI use <retrieveEIRequest> instead of the standard <retrieveFormInfoRequest>. retrieveEIRequest contains the payDayDate and submissionKey fields that are used to identify a return.

<**retrieveEIRequest**> structure:

```
<r:retrieveEIRequest xmlns:cmn="urn:www.ird.govt.nz/GWS:types/Common.v2"
                     xmlns:rc="urn:www.ird.govt.nz/GWS:types/ReturnCommon.v2"
                     xmlns:r="urn:www.ird.govt.nz/GWS:types/ReturnEI.v2"
                     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
                     xsi:schemaLocation="urn:www.ird.govt.nz/GWS:types/ReturnEI.v2">
        <cmn:softwareProviderData>
                <cmn:softwareProvider>SoftwareCompany</cmn:softwareProvider>
                <cmn:softwarePlatform>SoftwarePlatform</cmn:softwarePlatform>
                <cmn:softwareRelease>V1.1</cmn:softwareRelease>
        </cmn:softwareProviderData>
        <cmn:identifier IdentifierValueType="ACCIRD">123456789</cmn:identifier>
        <cmn:accountType>EMP</cmn:accountType>
        <rc:periodEndDate>2018-01-31</rc:periodEndDate>
        <rc:majorFormType>EI2</rc:majorFormType>
        <r:payDayDate>2018-01-01</r:payDayDate>
</r:retrieveEIRequest>
```

Field	Requirement	Description
payDayDate	Required	Payday means the day on which an employer makes a PAYE income payment to an employee

<**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></errorMessage>
        </statusMessage>
        <responseBody xmlns:r="urn:www.ird.govt.nz/GWS:types/ReturnEI.v2"
                      xsi:type="r:PrepopResponseBodyType">
                <r:accountId>123456798EMP001</r:accountId>
                <r:employee>
                        <r:irdNumber>123456789</r:irdNumber>
                        <r:employeeName>MarkTwain</r:name>
                        <r:taxCode>M</r:taxCode>
                        <r:employmentStartDate></r:employmentStartDate>
                        <r:employmentFinishDate></r:employmentEndDate>
                </r:employee>
                <r:employee>
                        <r:irdNumber>123456789</r:irdNumber>
                        <r:employeeName>SamualClemens</r:name>
                        <r:taxCode>M</r:taxCode>
                </r:employee>
        </r:responseBody>
</r:prepopResponse>
```

Field	Description
irdNumber	The IRD number of the employee—if no IRD number is available, 000000000 will be returned
employeeName	The name the employee is known as on the EI (255 characters)
taxCode	The tax code under which the employee falls
employmentStartDate	The start date of the employment (if date is not known, nothing will be returned)
employmentFinishDate	The cease date of the employment (if employment is still on-going, nothing will be returned)

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 EI Return Service import a common.xsd which has some data types specific to Inland Revenue. The 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 section 4 for where to find schemas for this service.

4.3 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

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

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 ReturnEI.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 ReturnEI.v2.xsd.

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

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.

NOTE: *These response codes are subject to change and additional codes may be added from time to time. Software consuming this service* **must** *be able to account for additional responses that may not have been specifically coded.*

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, IPS, NRT, PIE, PRS, PSO, EMP, RLT, RWT.
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
(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 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	Filing period does not exist (eg attempting to file for a filing period such as 17-Feb-2019)
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.
107	Duplicate return	There is already a return for this period/paydate (for EI). An amendment to a return that has already been submitted must be submitted with isAmended as true.

Standard codes	Standard message	Description
108	Return locked for processing	Return locked for processing—try again later
109	Invalid Amend Reason	When amending a return there must be a valid amend reason.
140	Invalid minor form type	Invalid minor form type
144	Return being submitted	The return has been submitted and queued for processing. The return will be processed in near real time.
145	Return held for processing	The return is not visible because it is in a non-amendable error.

5.3 Employment Information-specific response codes

The following response codes are specific to EI Return Service calls:

Standard codes	Standard message	Description
131	Duplicate line items	There were two line items with the same number in the same submission
132	Reverse/replace can only be used for an amendment	Customer tried to reverse/replace a return without amending the return
134	Invalid employee IRD number	A provided employee IRD number is invalid. **Note:** IRD number must pass modulus 11 digit check
150	Credit transfer requests are not supported	Credit transfer requests are not supported for the form type
160	Duplicate payday submission	A payday submission has already been submitted for the same account, period, paydate, and request payload in the last hour.
161	Payday date not in filing period	The payday date is not in the same month as the filing period (eg payday date 31/03/2019 and filing period 30/04/2019)
162	Period has non-payday return	The return filing period already has an EMS / IR348 filed.
163	Pay period end date before pay period start	The value of 'payPeriodEndDate' must be on/or after the 'payPeriodStartDate'.
164	Period too far into the future	The requested filing period is too far into the future and unavailable for filing at this time (maximum two months in advance)

Standard codes	Standard message	Description
167	Return is reversed	The return has been manually reversed and cannot be amended. The submission key is no longer valid.
168	Return is transferred	The return has been manually transferred to another account or period. If this is incorrect please contact Inland Revenue (either via Phone or MyIR)
169	Submitted incorrect EI version	The payday return was previously submitted as a different version of EI. • EI v1 cannot be used to amend or retrieve EI v2 • EI v2 cannot be used to amend or retrieve EI v1
171	Tax code unsupported EI version 2	The provided tax code is not supported **Note:** In v2 of EI, it is not possible to add an additional line item with tax code of either ESS, SLCIR, or SLBOR
173	Account was not active for the period submitted	Account was not active for the period submitted

6 Glossary

Acronym/term	Definition
ACC	Accident Compensation Corporation Account ID
ACCIRD	Account IRD
AIL	Approved Issue Levy
API	Application Programming Interface—a set of functions and procedures that allow applications to access the data or features of another application, operating system or other service.
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
DHCP	Dynamic Host Configuration Protocol—a client/server protocol that automatically provides an IP host with its IP address and other related configuration information such as the default gateway.
DWT	Dividend Withholding Tax
EI	Employment Information
End points	A term used to describe a web service that has been implemented
ESCT	Employer Superannuation Contribution Tax—one of the many deductions that come from payroll
FIPS	Federal Information Processing Standard—a suite of IT standards from the US Federal Government
Gateway	Inland Revenue's web services gateway
GWS	Gateway Services—the brand name for the suite of web services that Inland Revenue is providing. The EI Return 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.
IAMS	Identity and Access Management—a logical component that performs authentication and authorisation. Physically it is a set of discrete hardware and software products, plug-ins and protocols. Usually implemented as separate External IAMS (XIAMS) and Internal IAMS.
IAS	Identity and Access
INC	Inland Revenue's abbreviation for Income Tax
IP	Internet Protocol—the principal communication protocol in the Internet protocol suite for relaying datagrams across networks
IPS	Interest Pay-as-you-earn
IRD	Inland Revenue Department
MSD	NZ Ministry of Social Development

Acronym/term	Definition
OAuth	An HTTPS based protocol for authorising access to a resource, currently at version 2
PAYE	Pay As You Earn
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 an EI return) must/can contain, as well as validating the payload.
SHA	Secure Hashing Algorithm. There is a family of them that provide different strengths. SHA-2 is currently favoured over SHA-1, which has been compromised.
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.
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.
URL	Universal Resource Locator—also known as a web address
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 X.509 certificate to which it is bound identifies whose key it is, who issued it, when it expires etc. When a counterparty's X.509 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, trading partners etc, 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

Inland Revenue
Te Tari Taake

7 Change log

This table lists all changes that have been made to this build pack document since v2.0 was released.

Version	Date of change	Document section	Description
V2.0	25/08/20	V2 SERVICE UPDATES	• Removed comment that a return can take up to a day to process. • Added comment that returns will process within 5 minutes
	28/07/20	5.3	• New error code 173 added
		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.*
		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
	03/04/20		• V2 released