|  |  |  |  |
| --- | --- | --- | --- |
|  | | | |
|  | |  |  |
| Standard Business Reporting  *Interposed Entity Election or Revocation* Message Implementation Guide  Program name: Standard Business Reporting  Date: 05 August 2013  FOR PRODUCTION USE - suitable for use | | | |
|  | | | |
| * This document and its attachments are **Unclassified** |  | | |
|  | For further information or questions, contact the SBR Service Desk at [SBRServiceDesk@sbr.gov.au](mailto:SBRServiceDesk@sbr.gov.au) or call 1300 488 231. International callers may use +61-2-6216 5577 | | |

VERSION CONTROL

|  |  |  |
| --- | --- | --- |
| Version | Release date | Description of changes |
| 1.0 | 28/04/2011 | * Multiple declarations from Trustees and Partners: Included a new tuple RP:iee.0002.lodge.req.xx.xx:DeclarationOfTrusteeOrPartner (MIG Seq RP-23) containing tuples prsnstrcnm3.xx.xx:PersonNameDetails, orgname2.xx.xx:OrganisationNameDetails, and declaration2.xx.xx:Declaration * Added contact phone number details (MIG Seq INT-4) for a tax agent – INT:electroniccontacttelephone1.xx.xx:ElectronicContactTelephone |
| 1.1 | 24/06/2011 | Section 2.1.1- Lodgement with a parent return   * Updated the maximum number of IEE schedules that can be lodged with a parent return to 50.   Section 5.5.1.15 - IEE.LODGE Request Message Content Table   * VR.ATO.IEE.410266 rule logic updated (sequence number 3.1, 23.3.1, and 25.1) * Removed rule VR.ATO.IEE.410249 as it is not applicable to IEE10 (sequence number 6.3) * Corrected XBRL Fact from “pyde.xx.xx:OrganisationNameDetails.OrganisationalNameType.Code” to “pyde.xx.xx:OrganisationNameDetails.Currency.Code” (sequence number 21.5.2) * Removed “IEE standard organisation rules” reference and Ruleset:orgname2 for “pyde.xx.xx:OrganisationNameDetails.Currency.Code” (sequence number 21.5.2) * Removed rule VR.ATO.IEE.410268 as it is not applicable to IEE147 (sequence number 25.1) * Removed rule VR.ATO.IEE.410252 as it is not applicable to IEE192 (sequence number 25.2) * Added new rule VR.ATO.IEE.410264 for IEE192 (sequence number 25.2)   Appendix C - Validation rules alias definitions   * Alias definitions updated for IEE88,IEE146, IEE186 and IEE203 * New alias definition added for IEE192 |
| 1.2 | 05/08/2013 | Section 5.5.1.5 - IEE.LODGE Request Message Content Table   * SWS1073 Remove mandatory requirement of organisation tuple within TrusteeOrPartner tuple |

|  |  |  |
| --- | --- | --- |
| ENDORSEMENT  APPROVAL |  |  |
|  | Solutions Integration  Standard Business Reporting | |
| Michael Ferris | Project Manager  Strategic Web Services  Australian Taxation Office | |

Copyright

© Commonwealth of Australia 2013 (see exceptions below).   
This work is copyright.  Use of this Information and Material is subject to the terms and conditions in the "SBR Disclaimer and Conditions of Use" which is available at [http://www.sbr.gov.au](http://www.sbr.gov.au/). You must ensure that you comply with those terms and conditions. In particular, those terms and conditions include disclaimers and limitations on the liability of the Commonwealth and an indemnity from you to the Commonwealth and its personnel, the SBR Agencies and their personnel.   
    
You must include this copyright notice in all copies of this Information and Material which you create.  If you modify, adapt or prepare derivative works of the Information and Material, the notice must still be included but you must add your own copyright statement to your modification, adaptation or derivative work which makes clear the nature of your modification, adaptation or derivative work and you must include an acknowledgement that the adaptation, modification or derivative work is based on Commonwealth or SBR Agency owned Information and Material.  Copyright in SBR Agency specific aspects of the SBR Reporting Taxonomy is owned by the relevant SBR Agency.

TABLE OF CONTENTS

[Section 2.1.1- Lodgement with a parent return 2](#_Toc363202015)

[1 Introduction 6](#_Toc363202016)

[1.1 PURPOSE 6](#_Toc363202017)

[1.2 AUDIENCE AND SCOPE 6](#_Toc363202018)

[1.3 REFERENCES 6](#_Toc363202019)

[1.4 CHANGE MANAGEMENT 7](#_Toc363202020)

[2 General Instructions 8](#_Toc363202021)

[2.1 INCOME TAX SUITE OVERVIEW 8](#_Toc363202022)

[2.1.1 Lodgement with a parent return 8](#_Toc363202023)

[2.1.2 Message Structure 8](#_Toc363202024)

[2.1.3 Taxonomy and MIG structure 8](#_Toc363202025)

[2.1.4 Schema Use 8](#_Toc363202026)

[2.2 AUTHORISATION OF INTERMEDIARIES 9](#_Toc363202027)

[2.3 MONETARY AMOUNT 9](#_Toc363202028)

[2.4 DECLARATIONS 9](#_Toc363202029)

[2.4.1 Interposed Entity Election or Revocation (IEE) declaration 12](#_Toc363202030)

[2.5 SBDH VARIATIONS 13](#_Toc363202031)

[2.5.1 Business Documents 13](#_Toc363202032)

[2.5.2 Attachments 13](#_Toc363202033)

[2.5.3 Document identifiers 13](#_Toc363202034)

[2.6 RESPONSE MESSAGES 14](#_Toc363202035)

[2.6.1 Messages Described in the MIG 14](#_Toc363202036)

[2.7 VALIDATION PHASING 14](#_Toc363202037)

[2.8 RULE EXPRESSION 14](#_Toc363202038)

[2.8.1 Context MIG Labels 14](#_Toc363202039)

[2.8.2 Form Prefix Labels 14](#_Toc363202040)

[2.8.3 No Form or Context Prefix 15](#_Toc363202041)

[2.8.4 Use of xx.xx in fact names 15](#_Toc363202042)

[2.8.5 Use of Aliases 15](#_Toc363202043)

[2.8.6 Interpretation of NULL in calculations and comparisons 15](#_Toc363202044)

[2.8.7 Case Sensitivity 15](#_Toc363202045)

[2.9 TUPLES AND CONTEXT 15](#_Toc363202046)

[2.10 FORMAT RULES 15](#_Toc363202047)

[3 Business overview 17](#_Toc363202048)

[3.1 BUSINESS CONTEXT MODEL 17](#_Toc363202049)

[3.2 FINANCIAL YEAR AND SUBSTITUTED ACCOUNTING PERIODS 19](#_Toc363202050)

[3.3 BUSINESS APPLICABILITY PERIOD 19](#_Toc363202051)

[3.4 Report version 20](#_Toc363202052)

[4 XBRL context specification 21](#_Toc363202053)

[4.1 CONTEXT SPECIFICATION – DIMENSION 1: ReportPartyTypeDimension 21](#_Toc363202054)

[4.1.1 Context instances 22](#_Toc363202055)

[5 Interaction model 24](#_Toc363202056)

[5.1 PREREQUISITES 24](#_Toc363202057)

[5.2 SERVICE SUMMARY 25](#_Toc363202058)

[5.3 IEE FORM BUSINESS DOCUMENT INSTANCE LIMITATION 25](#_Toc363202059)

[5.4 IEE.PRELODGE SPECIFICATIONS 26](#_Toc363202060)

[5.4.1 IEE.PRELODGE Request - Message 26](#_Toc363202061)

[5.4.2 IEE.PRELODGE Response - Message 27](#_Toc363202062)

[5.5 IEE.LODGE SPECIFICATIONS 27](#_Toc363202063)

[5.5.1 IEE.LODGE Request - Message 28](#_Toc363202064)

[5.5.2 IEE.LODGE Response - Message 60](#_Toc363202065)

[Appendix A – The message content table explained 61](#_Toc363202066)

[Appendix B – Tax Office Structured English 64](#_Toc363202067)

[Appendix C – Validation rules alias definitions 69](#_Toc363202068)

[Appendix D – Differences between the IEE standalone form and the IEE schedule 73](#_Toc363202069)

[Appendix E – Common Validation Rules 74](#_Toc363202070)

[Appendix F – Domain Definitions 81](#_Toc363202071)

Terminology

For definition of the terminology and acronyms used within this document please refer to the glossary on the SBR website – Click here [**http://www.sbr.gov.au/software-developers/developer-tools/glossary**](http://www.sbr.gov.au/software-developers/developer-tools/glossary)to go to the glossary.

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in RFC 2119 **http://www.ietf.org/rfc/rfc2119.txt**. The use of the word “Mandatory” is to be read as “MUST”.

1. Introduction
   1. PURPOSE

The purpose of this document is to support the software developer in the implementation of the SBR Interposed Entity Election or Revocation (IEE) reporting service and reporting taxonomy for the Australian Taxation Office (ATO).

The IEE may be lodged as a standalone form or it may be lodged as a schedule with any of the following ITR forms:

* Company Return (CTR)
* Partnership Return (PTR)
* Trust Return (TRT); or
* SMSF Annual Return (SMSFAR)

Where an IEE accompanies an ITR form, refer to that ‘parent’ MIG for details of the relevant service.

The current version of the IEE only supports Company entities. Support for other entities will follow in future releases.

* 1. AUDIENCE AND SCOPE

This document contains the necessary information required to support the ATO IEE implementation.

* 1. REFERENCES

|  |  |  |
| --- | --- | --- |
| **Ref** | **Document Link** | **Document description** |
|  | The SBR Web Service Implementation Guide document can be downloaded at  [**http://www.sbr.gov.au/software-developers/developer-tools/web-services**](http://www.sbr.gov.au/software-developers/developer-tools/web-services) | Technical interface data that is common to all business processes and messages that use the SBR channel:   * Web service protocol specifications * Standard message header structure * Standard error codes * Authentication protocol and trust broker |
|  | The SBR Taxonomy Architecture document can be downloaded at  [**http://www.sbr.gov.au/software-developers/developer-tools/re-usable-components**](http://www.sbr.gov.au/software-developers/developer-tools/re-usable-components) | Reference document that describes the structure of the SBR taxonomy, its naming conventions, release management and change control, and how each business interaction fits within the architecture. |
|  | The Software Developer Kit documentation can be accessed at  [**http://www.sbr.gov.au/software-developers/enabling-sbr-in-my-application/productivity-tools**](http://www.sbr.gov.au/software-developers/enabling-sbr-in-my-application/productivity-tools) | Reference information for the software developer using the SBR software developer kit. |

* 1. CHANGE MANAGEMENT

If a material change is required to the ATO IEE Message Implementation Guide (MIG) the document will be re-released. The Taxonomy Approval Committee must approve any change.

1. General Instructions

This section provides instructions that are relevant across all collaborations and messages specified within this MIG.

* 1. INCOME TAX SUITE OVERVIEW
     1. Lodgement with a parent return

The IEE is part of the ATO SBR Income Tax Suite. This suite of reporting taxonomies combine to allow a taxpayer (Reporting Party) to lodge an ITR.

When lodging an Income Tax Return a Reporting Party must populate a return document (e.g. CTR) and, dependent on their business activities, may need to lodge one or more schedules (e.g. Capital Gains Tax Schedule) with the return.

Alternatively, some schedules may be lodged as a standalone via specific services for lodgment outside of an Income Tax Return.

The table below shows which Income Tax Returns an IEE may be submitted with, however the circumstances in which the schedule is to be provided are detailed in the validation rules.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Schedule | Company Return  (CTR) | Partnership Return  (PTR) | Trust Return (TRT) | SMSF Annual Return (SMSFAR) |
| Interposed Entity Election (IEE) | 0..50 | 0..50 | 0..50 | 0..50 |

* + 1. Message Structure

Where an Income Tax Return is lodged with an associated schedule, they will be all included in one SBR message. The return and each schedule will be separate Business Documents (XBRL instances) within the Standard Business Document Body structure as defined in the WIG.

* + 1. Taxonomy and MIG structure

Within the Income Tax suite each return and schedule will have its own reporting taxonomy and MIG. To enable lodgment of an Income Tax Return a software developer will need to consider these taxonomies and MIGs together.

Where a schedule can only be lodged in conjunction with a return, the details in the MIG for that schedule will only cover information relating to the schema for that schedule. The MIG for the return will contain the details of the interaction and the overall message.

* + 1. Schema Use

Please note that the lodge schema for this report will be used for both the pre-lodge and lodge interactions.

* 1. AUTHORISATION OF INTERMEDIARIES

For all IEE request messages, which are sent in as a stand-alone form, the ATO will check against its records that the sender is authorised to perform the requested action for the reporting party (taxpayer).

The checking will compare the identity and permissions associated with the AUSkey against the identity provided in the business document for the reporting party. If the sender is acting in their role as an agent, then the Tax Agent Number must also be provided otherwise authorisation will fail.

If authorisation fails, then a response message communicating this in the Standard Business Document Header Message Event section will be returned with SBR Error Code: CMN.ATO.GEN.200000.

* 1. MONETARY AMOUNT

All data elements of type xbrli:MonetaryItemType supplied within messages associate to this report are required to be in Australian dollars. In adherence with the XBRL standard this is achieved by using the following unit declaration:

<xbrli:unit id="u1">

    <xbrli:measure>iso4217:AUD</xbrli:measure>

</xbrli:unit>

XBRL processors will validate that the measure adheres to the ISO standard but the agency will ensure that this is set to Australian Dollars.

* 1. DECLARATIONS

Where the IEE is sent in as a stand-alone form, it requires a declaration from the reporting party (taxpayer) indicating that the information contained is true and correct. It may also be lodged by an agent making a different declaration.

To make a declaration, the sender needs to be cognisant of two things:

* The statement they are making, and
* That it becomes a declaration by them ‘signing’ it.

As a result, in every case that a declaration is required to accompany a transaction, the sender must have displayed to them:

* Specific statement(s) describing what they are about to declare, and
* An acknowledgement that the declaration is being made by signing the statement(s) in a particular way.

The sender signs by actively confirming what constitutes their ‘signature’ using a tick-box, submit button, or similar mechanism. Their signature must be some information sent with the transaction that enables the sender to be uniquely identified within the business.

How this declaration is made and handled via SBR varies depending on who sends the form and what type of AUSkey they use to do so. The instructions for each case are described in the tables below.

The IEE only requires a declaration to be made if it is sent as a stand-alone form. It does not require a separate declaration to be made if it is sent in accompanying an ITR form. In this latter case the declaration is made as described in the ‘parent’ form MIG.

|  |  |
| --- | --- |
| Case 1: A **reporting party** (ie. taxpayer), or an intermediary who is not an agent, is lodging using an AUSkey assigned to an **individual**. | |
| Declaration statement | The statement that the sender is declaring shall be:  “*I declare that the information transmitted in this Interposed Entity Election or Revocation Form is true and correct and that I am authorised to make this declaration. The form requirements have been met*.” |
| Signing statement | The text describing the way that they are ‘making’ the declaration by ‘signing’ it in a particular way shall include reference to signing with the AUSkey.  For example: “*Tick this box to sign this declaration with the AUSkey you used to log in*.”  A statement “Tick this box to sign this declaration” would not be acceptable as it does not state the identity the sender is using to make the declaration. |

|  |  |
| --- | --- |
| Case 2: A **reporting party** (ie. taxpayer), or an intermediary who is not an agent, is lodging via SBR using an AUSkey assigned to a **device**. | |
| Declaration statement | The statement that the sender is declaring shall be:  “*I declare that the information transmitted in this Interposed Entity Election or Revocation is true and correct and that I am authorised to make this declaration.*” |
| Signing statement | The text describing the way that they are ‘making’ the declaration by ‘signing’ it in a particular way shall include reference to signing with the AUSkey for the device and the field giving the unique user identifier.  For example: “*Tick this box to sign this declaration with the AUSkey you used to log in and your full name inserted above*.”  A statement “Tick this box to sign this declaration” would not be acceptable as it does not state the identity the sender is using to make the declaration.  The user identifier must allow the AUSkey owner or an external auditor to uniquely identify the individual who made the declaration.  The identifier used can be specified by the AUSkey owner providing it allows identification as mentioned above. Examples of suitable identifiers include a user login, a full name, or an email address. |

|  |  |
| --- | --- |
| Case 3: An **intermediary** who is an agent (ie. Tax agent or BAS service provider) is lodging via SBR using an AUSkey assigned to an **individual**. | |
| Declaration statement | The statement that the sender is declaring shall be:  “*I declare that:*   * *I have prepared this Interposed Entity Election or Revocation in accordance with the information supplied by the entity;* * *I have received a declaration made by the entity that the information provided to me for the preparation of this form is true and correct; and* * *I am authorised by the entity to give information in this form to the Commissioner.”* |
| Signing statement | The text describing the way that they are ‘making’ the declaration by ‘signing’ it in a particular way shall include reference to signing with the AUSkey.  For example: “*Tick this box to sign this declaration with the AUSkey you used to log in.”*  A statement “Tick this box to sign this declaration” would not be acceptable as it does not state the identity the sender is using to make the declaration. |

|  |  |
| --- | --- |
| Case 4: An **intermediary** who is an agent (ie. Tax agent or BAS service provider) is lodging via SBR using an AUSkey assigned to a **device**. | |
| Declaration statement | The statement that the sender is declaring shall be:  “*I declare that:*   * *I have prepared this Interposed Entity Election or Revocation in accordance with the information supplied by the entity;* * *I have received a declaration made by the entity that the information provided to me for the preparation of this form is true and correct; and* * *I am authorised by the entity to give information in this form to the Commissioner.”* |
| Signing statement | The text describing the way that they are ‘making’ the declaration by ‘signing’ it in a particular way shall include reference to signing with the AUSkey for the device *and* the field giving a unique user identifier.  For example: “*Tick this box to sign this declaration with the AUSkey used by this software and your full name inserted above.””*  A statement “Tick this box to sign this declaration” would not be acceptable as it does not state the identity the sender is using to make the declaration.  The user identifier must allow the AUSkey owner or an external auditor to uniquely identify the individual who made the declaration.  The identifier used can be specified by the AUSkey owner providing it allows identification as mentioned above. Examples of suitable identifiers include a user login, a full name, or an email address. |

* + 1. Interposed Entity Election or Revocation (IEE) declaration

IEE can be submitted as a schedule with a parent form (Company, Trust, Partnership and Self Managed Superannuation Fund) or on its own as a standalone form.

* + - 1. Declaration for IEE schedule:

IEE as a schedule must have one of these declarations irrespective of the sender.

If an IEE schedule is submitted with a Company Tax Return (CTR) then the declaration is made by the company’s public officer. The Reporting Party declaration must be completed.

If an IEE schedule is submitted with a Trust Return (TRT) or Partnership Return (PTR) or Self Managed Superannuation Fund Annual Return (SMSFAR) then the declaration is made by the trustee or partner. The “Declaration of Trustee or Partner” tuple (includes person name details, organisation name details and declaration) must be completed.

**Please note** that there can be up to 10 Trustees/Partners (Trustee or Partner tuple) and must have same number of trustee/partner declarations. At least one trustee/partner declaration must be completed as “TrueAndCorrect”.

If the parent return (CTR, TRT, PTR, and SMSFAR) is submitted by an Intermediary, then Intermediary declaration must not be completed for IEE schedule.

**Parent Return:**

**CTR** – Company’s public officer – Reporting Party (RP) Declaration

**TRT/PTR/SMSFAR** – Trustee or Partner –DeclarationOfTrusteeOrPartner

**Intermediary (Sender) for Parent Return** – No Intermediary (INT) Declaration for IEE schedule

* + - 1. Declaration for IEE standalone:

IEE standalone form must have one of these declarations depending on the sender.

**IEE form submitted by Self Lodger:**

If the IEE form is submitted by a self lodger (only Reporting Party context) then declaration must be completed based on the Interposed entity election - Entity code.

If the entity code =’051’– Company, then the company’s public officer (Reporting Party) declaration must be completed.

If the entity code =’161’- Partnership, or ’212’- Trust, or ’217’ – Self Managed Superannuation Fund, then the Declaration of Trustee or Partner tuple (includes person name details, organisation name details and declaration) must be completed. As mentioned earlier that there must be equal number of trustee/partner declaration to the trustee or partner tuple with one declaration completed as “TrueAndCorrect”.

**IEE form submitted by Intermediary (Business Cross entity & Tax Agent):**

If the IEE form is submitted by an Intermediary (Reporting Party and Intermediary context) then only Intermediary (sender) declaration must be completed irrespective of the Interposed entity election - Entity code.

If the entity code =’051’– Company or ’161’- Partnership, or ’212’- Trust, or ’217’ – Self Managed Super Fund, only Intermediary declaration must be completed.

**Sender: Reporting Party (RP)**

**Entity type – Company (051) -** Company’s public officer – Reporting Party Declaration

**Entity type – Partnership (161)** – Trustee or Partner – DeclarationOfTrusteeOrPartner

**Entity type – Trust (212)** – Trustee or Partner – DeclarationOfTrusteeOrPartner

**Entity type – Self Managed Superannuation Fund (217)** – Trustee or Partner – DeclarationOfTrusteeOrPartner

**Sender: Intermediary (INT)**

**Entity type – Company (051) –** Intermediary (sender) declaration

**Entity type – Partnership (161)** – Intermediary (sender) declaration

**Entity type – Trust (212)** – Intermediary (sender) declaration

**Entity type – Self Managed Superannuation Fund (217)** – Intermediary (sender) declaration.

* 1. SBDH VARIATIONS

The Web Services Implementation Guide (WIG) describes the Standard Business Document Header (SBDH) content in detail. Described in this section are only variations from what is defined in the WIG. Where the Interposed Entity Election or Revocation is sent in accompanying one of the ITR forms, refer to that ‘parent’ MIG for details on SBDH variations.

* + 1. Business Documents

Only one business document in the SBDB per message will be accepted for IEE messages where the form is sent in as a stand-alone form. Where the IEE is sent in as a schedule accompanying an ITR form, more than one IEE may be sent - refer to MIGs describing the parent returns for details.

* + 1. Attachments

No attachments will be accepted or provided for IEE interactions where the form is sent in as a stand-alone form. Where it is sent in accompanying an ITR form, refer to MIGs describing the parent returns for details.

* + 1. Document identifiers

The sbdm:BusinessDocument.GovernmentGeneratedIdentifier.Text field will not be used for IEE interactions where it is sent in as a stand-alone form. Where it is sent in as a schedule accompanying an ITR form, refer to MIGs describing the parent returns for details.

* 1. RESPONSE MESSAGES
     1. Messages Described in the MIG

Where business rules associated with data elements could be reasonably implemented by a Software Developer they have been described in the business content tables below along with an associated response message code. A description of response message coding can be found in Appendix A of this document.

If there is mention of a label or a section in the error message, it is referring to the label or section as displayed on the paper form (PDF versions are available at [www.ato.gov.au](http://www.ato.gov.au/)).

* 1. VALIDATION PHASING

The validation rules described in this document will be applied to the Business Document in phases. Validation will not progress to the next phase until the current phase is completely passed. This is implemented so as to avoid multiple unecessary messages being returned. As an example, if an IEE business document was not provided this would be an error. If a phased validation approach were not used, then potentially an error for each missing of the mandatory fields could be returned as well.

The phases implemented in the Income Tax suite will be as follows:

1. SBDH checks
2. XBRL contexts, Formats, Data types, lengths and enumerations
3. Presence of mandatory fields
4. Cross field rules, calculations, comparisons
5. Cross form (cross Business Document) rules
   1. RULE EXPRESSION

Many of the rules in the tables below are written in Tax Office Structured English. This is a type of pseudo code and has been used to ensure clarity in rule expression. For explanations of terms used in Tax Office Structured English see Appendix B.

* + 1. Context MIG Labels

Context Instance labels will be given to each possible instance of an XBRL context and will be used within the MIG to describe the context and link a fact to a context. For example, a fact may appear in a business rule with a prefix CTR:RP.TOFA:bafpr1.xx.xx:Income.FinancialArrangementsUnrealisedGains.Amount

This indicates this fact is being reported in the context where the dimension ReportPartyType is set to “ReportingParty” and the dimension FinancialArrangementType is set to “TOFA”.

* + 1. Form Prefix Labels

Due to multiple business documents existing in the one ITR message (ie. The ITR and any of the listed schedules), cross-form rules must be applied. To ensure it is clear which business document an XBRL fact is from, a form pre-fix is included in any fact description.

For example: IEE:RP:pyid.xx.xx:Identifiers.AustralianBusinessNumber.Identifier means this field is in the IEE form.

Where the IEE is lodged with an ITR return, the ITR return will be referenced as the “PARENT RETURN”.

* + 1. No Form or Context Prefix

Where no form or context prefix (as described above) is applied to a fact within a rule, it is because the rule is reused on multiple forms or it should be applied regardless of context within the form.

* + 1. Use of xx.xx in fact names

In the actual Business Document, an XBRL fact will have a namespace prefix including the version of the element, for example bafpr.02.00:Income.FinancialArrangementsUnrealisedGains.Amount

For the purposes of the MIG, the version contained within the prefix has been replaced with xx.xx. The correct version can be derived from the Discoverable Taxonomy Set.

* + 1. Use of Aliases

In order to make the validation rules more readable aliases have been used in some rules. An alias is enclosed in square brackets eg: [IEE123]. A full list of aliases used in this MIG and their XBRL definition is enclosed in Appendix C.

* + 1. Interpretation of NULL in calculations and comparisons

Where a rule involves a calculation or a comparison with a number, we will consider NULL (xsi:nil=true) or absent XBRL facts as zero for the purposes of the calculation or comparison.

* + 1. Case Sensitivity

Many rules contain a comparison with a string value enclosed in inverted commas e.g. “Australia”. The case used in these comparisons reflects the most common usage, however any comparisons are case **in**sensitive. That means that the test: IF <a> = “Australia” would be true if <a> were “australia”, “AUSTRALIA”, “Australia” or “AuStRaLiA”.

* 1. TUPLES AND CONTEXT

All facts reported within a tuple instance (including nested tuples) must use the same context.

* 1. FORMAT RULES

There are format rules that will be applied to facts provided in request documents. Given that they are heavily reused and very simple, they have been included here rather than provided in the message tables below for each fact.

| **XBRL data type** | **Rules** | **Rule Imp.** | **SBR Msg Code** |
| --- | --- | --- | --- |
| Any | Data supplied longer than maximum field length | See Schematron Rules against each individual field | CMN.ATO.GEN.001011 |
| xbrli:dateItemType | Date invalid or format incorrect | Schematron ID = VR.ATO.GEN.001008 | CMN.ATO.GEN.001008 |
| xbrli:monetaryItemType | Monetary format incorrect | Schematron ID = VR.ATO.GEN.001009 | CMN.ATO.GEN.001009 |
| xbrli:booleanItemType | Boolean format incorrect | Schematron ID = VR.ATO.GEN.001010 | CMN.ATO.GEN.001010 |
| ID | ID format incorrect | Schematron ID = VR.ATO.GEN.001012 | CMN.ATO.GEN.001012 |
| xbrli:positiveIntegerItemType | Positive integer format incorrect | Schematron ID = VR.ATO.GEN.001013 | CMN.ATO.GEN.001013 |
| xbrli:nonNegativeIntegerItemType | Non negative integer format incorrect | Schematron ID = VR.ATO.GEN.001014 | CMN.ATO.GEN.001014 |
| xbrli:pureItemType | Pure format incorrect | Schematron ID = VR.ATO.GEN.001015 | CMN.ATO.GEN.001015 |
| Any | Data supplied does not match the xsd:pattern value | Schematron ID = VR.ATO.GEN.001017 | CMN.ATO.GEN.001017 |
| xbrli:decimalItemType | Decimal format incorrect | Schematron ID = VR.ATO.GEN.001018 | CMN.ATO.GEN.001018 |

1. Business overview
   1. BUSINESS CONTEXT MODEL

This section provides a high level context for Income Tax Return interactions – including IEE - between a business, or an intermediary, and the ATO for the purpose of meeting an Income Tax obligation. The interactions are broadly divided into five phases:

* *Register and maintain*. Obtaining an AUSkey from the Australian Business Register (ABR) to be used to authenticate interactions. Intermediaries also need to register with the ATO and create an intermediary relationship with any taxpayers they will lodge for via SBR. These interactions are out of scope for SBR web services implementation at this time. They are not described in this document however have been included for context.
* *Notify obligations.* The ATO notifies a business of an impending obligation. This interaction is out of scope for SBR implementation. It is not described in this document however has been included for context.
* *Prepare and lodge interactions*. The ATO provides, via SBR, services to validate (prelodge) an Income Tax Return – which may include an Interposed Entity Election or Revocation form - before lodging, and the lodgment of the form. These interactions are in scope for SBR implementation and are described in detail in this document.
* *Post lodgment phase*. The ATO notifies the business/intermediary of the result of the income tax assessment. The business or intermediary may also enquire as to the progress of an income tax assessment. This interaction is out of scope for SBR implementation. It is not described in this document however has been included for context.
* *Payment interactions*. Businesses make payments or receive refunds determined by their Income Tax assessment. The payments are made through various mechanisms. These payment services are out of scope for SBR implementation and are not described in detail in this document however have been included for context.

Each of the phases described above could be performed by a number of different channels such as paper, Business portal, Tax agent portal, ELS and Web services. In the context of Interposed Entity Election or Revocation interactions for this document, it is the third phase (Prepare and Lodge) which is being enabled via the SBR channel. The remaining phases are out of scope of this document since they are performed by other existing ATO channels or the ABR.

Details of legislative requirements and other background on Income Tax returns are available at [www.ato.gov.au](http://www.ato.gov.au).



Figure 1: Describes the high level business context of an Income Tax Return interaction.

The following table lists each of the interactions shown above and describes the responsibilities for each in relation to implementation in SBR.

| **Interaction** | **Responsibility** |
| --- | --- |
| **1. Register and maintain** | The sender registers their identity (ABN/TFN) with the ABR and the ATO. Registration will include:   * Australian Business Number * Address details * Responsible person and authorisation for any intermediary * AUSkey   The sender sets up their preferences for dealing with the ATO such as method of submission and bank details.  Any changes to tax obligations or updates to their details are done by contacting the ATO or the ABR. |
| **2. Notify obligations** | The ATO will send out letters advising business when they need to lodge their ITR and any associated schedules.  Business may be able to arrange a deferral of time to lodge by writing to the ATO before the due date. |
| **3. Prepare and lodge interactions** | 1. The sender may conduct a pre-lodgment validation check on the form via SBR. 2. The sender lodges the Income Tax return and/or schedules.   The ATO validates the lodgment, and returns the results in a message back to the sender. |
| **4. Post Lodgment Phase** | The Tax office will process the IEE lodgment with the Income Tax return.  The ATO will issue Notice of Assessment/ Notice of Assessment (Amended) to business depending on the business type and if there is a payment or refund.  The business/intermediary may wish to contact the ATO to enquire the status of their ITR lodgment. |
| **5. Payment interactions** | Sender is required to pay the tax payable amount to the ATO by the payment due date unless other arrangements are made.  If the lodgment results in a refund, business/intermediary will receive a refund cheque via a NOA / NOAA. |

* 1. FINANCIAL YEAR AND SUBSTITUTED ACCOUNTING PERIODS

Most Reporting Parties will report Income Tax over the standard Australian financial year (1st July to 30th June). Some Reporting Parties will have a specific arrangement with the Tax Office to report Income Tax over a differenct financial year period called a Substituted Accounting Period (SAP). If the Reporting Party operates on a SAP, wherever a duration is specified for the report (for example in the RP context instance), the **SAP** start and end dates must be supplied.

* 1. BUSINESS APPLICABILITY PERIOD

The business applicability period for iee.0002 is the 2011, 2012 and 2013 financial year.

In SBR, there may not be a new schema released each business applicability period as with the paper form version. An expiry date will not be specified as the reporting taxonomy will continue to be valid for that income year in the future. Within an income year’s reporting taxonomy, if a schema is versioned (a new one is released in production) the previous schema will be supported for an appropriate transition period.

* 1. Report version

The SBR report version for IDS 2013 is **iee.0002.02.00**

1. XBRL context specification

The following sections define the context specification that will be used within this MIG. The context types are allocated to the individual data elements within the message specifications below.

* 1. CONTEXT SPECIFICATION – DIMENSION 1: ReportPartyTypeDimension

| **XBRL Instance Context Data Concept** | **Requirement** | **Instructions/Rules** | **Rule Implementation** | **SBR Message Code** |
| --- | --- | --- | --- | --- |
| Context Identifier | Mandatory | This is a unique identifier used to link the data element to a defined XBRL context. SBR is recommending a four character id starting with ‘C’ and a three digit sequential number for each context eg: C001  1. IF context id = NULLORBLANK  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.GEN.000240 | 1. CMN.ATO.GEN.200007 |
| Entity Identifier | Mandatory | This field must be set to the ABN or TFN of the entity that the part of the business document instance relates to. |  |  |
| Entity Identifier Scheme | Mandatory | This field must be set to **http://www.ato.gov.au/abn** or **http://www.ato.gov.au/tfn**. TFN must be used for the reporting party and ABN must be used for the Intermediary. | See below | See below |
| Entity Segment | Mandatory | Explicit member dimension ReportPartyType set to ReportingParty or Intermediary   1. IF (RprtPyType.xx.xx:ReportPartyTypeDimension <> "RprtPyType.02.03:Intermediary") AND (RprtPyType.xx.xx:ReportPartyTypeDimension <> "RprtPyType.02.03:ReportingParty")   RETURN VALIDATION MESSAGE  ENDIF  2. IF (Identifier Scheme <> "http://www.ato.gov.au/abn") AND (Identifier Scheme <> "http://www.ato.gov.au/tfn")  RETURN VALIDATION MESSAGE  ENDIF | 1. Schematron ID = VR.ATO.GEN.410196  2. Schematron ID = VR.ATO.GEN.000239 | 1. CMN.ATO.GEN.200007  2. CMN.ATO.GEN.200007 |
| Period Start Date | Mandatory | The start date must describe the start of the period that the return applies to. If the company operates on a standard financial year, this date would be set to 2010-07-01 for a 2011 tax return. If the company operates on a Substitute Accounting Period (SAP), this date must be set to the start of the SAP.  1. IF (period.startDate = NULLORBLANK WHERE CONTEXT(ALL))  RETURN VALIDATION MESSAGE ENDIF  2. IF (period.startDate >= period.endDate WHERE CONTEXT(ALL))  RETURN VALIDATION MESSAGE END IF | 1. Schematron ID = VR.ATO.GEN.000199  2. Schematron ID = VR.ATO.GEN.000201 | 1. CMN.ATO.GEN.001001  2. CMN.ATO.GEN.200009 |
| Period End Date | Mandatory | The end date must describe the end of the period that the return applies to. If the company operates on a standard financial year, this date must be set to 2011-06-30 for a 2011 tax return. If the company operates on a Substitute Accounting Period (SAP), this date must be set to the end of the SAP.  1. IF (period.endDate = NULLORBLANK WHERE CONTEXT(ALL))  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.GEN.000237 | 1.CMN.ATO.GEN.001001 |

* + 1. Context instances

| **Context instance MIG label** | **Dimensions with contrained values** | **Hypercube Name(s)** | **Instructions / Rules** | **Rule Implementation** | **SBR Message Code** |
| --- | --- | --- | --- | --- | --- |
| RP | “RprtPyType.02.03:ReportingParty” | ReportingParty  ReportingPartyOrIntermediary | 1. IF (RP:entity.identifier.TFN = NULLORBLANK)  RETURN VALIDATION MESSAGE ENDIF  2. WHERE PARENT RETURN EXISTS  IF (IEE:RP:entity.identifier.TFN <> NULLORBLANK) AND (IEE:RP:entity.identifier.TFN <> PARENT RETURN:RP:entity.identifier.TFN)  RETURN VALIDATION MESSAGE ENDIF  3 IF (RP:entity.identifier.TFN <> NULLORBLANK) AND (TFNALGORITHM (RP:entity.identifier.TFN) = FALSE)  RETURN VALIDATION MESSAGE  ENDIF  4. IF COUNT(RP) <> 1  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.GEN.410113  2. Schematron ID = VR.ATO.GEN.410115  3. Schematron ID= VR.ATO.GEN.428016  4. Schematron ID = VR.ATO.GEN.000208 | 1. CMN.ATO.GEN.410113  2. CMN.ATO.GEN.410115  3. CMN.ATO.GEN.428016  4. CMN.ATO.GEN.200007 |
| INT | “RprtPyType.02.03:Intermediary” | Intermediary  ReportingPartyOrIntermediary | 1. WHERE PARENT RETURN EXISTS IF (IEE:INT:entity.identifier.ABN <> NULLORBLANK) AND (IEE:INT:entity.identifier.ABN <> PARENT RETURN:INT:entity.identifier.ABN)  RETURN VALIDATION MESSAGE ENDIF 2. IF (entity.identifier.ABN) <> NULLORBLANK) AND (ABNALGORITHM (entity.identifier.ABN) = FALSE)  RETURN VALIDATION MESSAGE ENDIF 3. IF COUNT (INT) > 1  RETURN VALIDATION MESSAGE ENDIF 4. IF CONTEXT(INT) <> NULL AND ((INT:Identifier Scheme <> “http://www.ato.gov.au/abn”) OR (INT:entity.identifier.ABN = NULLORBLANK))  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.GEN.410119  2. Schematron ID = VR.ATO.GEN.430318  3. Schematron ID = VR.ATO.GEN.000226  4. Schematron ID = VR.ATO.GEN.410220 | 1. CMN.ATO.GEN.410119  1. CMN.ATO.GEN.430318  3. CMN.ATO.GEN.200007  4. CMN.ATO.GEN.000004 |

1. Interaction model



Figure 2: SBR interaction model for Income Tax Returns – including the IEE

* 1. PREREQUISITES

Prior to performing any of the SBR IEE interactions described in this document, the sender must have:

* Registered as an Australian business and obtained an ABN;
* Registered for business roles (for eg: ITR role) with the ABR or ATO;
* Obtained an AUSkey from the ABR; and
* Registered people with ‘User’ level SBR credentials to perform ITR tasks with the ATO or ABR. ‘Admin’ level SBR credentials will be authorised for any task for that ABN.

If an intermediary is lodging on behalf of a business, then the intermediary must have:

* Registered as an Australian business and obtained an ABN;
* Obtained an AUSkey from the ABR; and
* Registered an ‘intermediary-taxpayer’ relationship with the ATO for ITR lodgments, or had the taxpayer business do so.

On submitting a request via SBR, the ATO will verify that the sender is authorised to perform that task for the business that is reporting the Interposed Entity Election or Revocation.

If lodging via an intermediary, the business may not need an ABN unless their activities (eg: GST) require them to have one.

* 1. SERVICE SUMMARY

The IEE (NAT 2788) may be transmitted using SBR either as a stand-alone form or as an accompanying schedule to the Company Return (CTR) (NAT 0656), or Partnership Return (PTR) (NAT 0659), or Trust Return (TRT) (NAT 0660), or SMSF Annual Return (NAT 71226).

The interaction model defined above represents the abstract process model for Income Tax Returns – including IEE as a stand-alone form or as an accompanying schedule. The following defines the sequence, optionality and repeatability of the service execution. Where a service is identified as *‘Mandatory’* then the software developer has to implement this service as the service needs to be executed by the business in order to complete the transaction. Repeatability indicates if the service can be executed more than once – *‘Yes’* meaning that a service can be executed multiple times within the sequence.

|  | **Service name & description** | **Mandatory** | **Repeatability** |
| --- | --- | --- | --- |
| 1 | **prelodge**  The purpose of this transaction is to request pre-submission validation of an Interposed Entity Election or Revocation, where it is sent in as a stand-alone form, without processing the form. | No | Yes |
| 2 | **lodge**  The purpose of this transaction is to allow a client to lodge an Interposed Entity Election or Revocation, where it is sent in as a stand-alone form for processing. | Yes | No |

* 1. IEE FORM BUSINESS DOCUMENT INSTANCE LIMITATION

When lodged with an ITR form, the income tax suite allows for up to 50 IEE forms to be submitted.

In order for the Tax Office to provide a timely validation response to a request it is necessary to limit the number of IEE forms within an ITR lodgment. Initially the number of IEE forms must be limited to 50. If more than 50 IEE forms are provided, the following SBR response message code will be returned: CMN.ATO.CTR. 428011.

When lodged as a standalone form, only one IEE form may be lodged per submission.

* 1. IEE.PRELODGE SPECIFICATIONS

|  |  |
| --- | --- |
| **Interaction Name** | iee.prelodge |
| **Description** | The purpose of this transaction is to request pre-submission validation of an Interposed Entity Election or Revocation, where it is sent in as a stand-alone form, without processing the form. This transaction is optional. |
| **Stakeholders** | Reporting party, Intermediary, ATO |
| **Pre-conditions** | See Pre-conditions for iee.lodge |
| **Post-conditions** | For a successful pre-lodgement validation, the ATO will return:  A message event item informing successful validation.  For an unsuccessful pre-lodgement validation the ATO will return:  A message event item(s) containing a list of errors.  The Reporting party/Intermediary is then able to correct their form for re-submission. |
| **Initiating party** | Reporting party, Intermediary |
| **Channel** | SBR |
| **Core Service Map** | Prelodge |

* + 1. IEE.PRELODGE Request - Message
       1. Discoverable Taxonomy Set References

Refer to IEE.LODGE request – message.

* + - 1. Standard Business Document Header Content

##### The WIG provides the specification of the SBDH. The following table specifies the message-specific data element values or any variations to the WIG for the IEE.

| **Attribute Name** | **Instructions/Rules** | **Rule Imp** | **SBR Msg Code** |
| --- | --- | --- | --- |
| sbdm:Message.Type.Text | 1. Mandatory - Value must be “iee.0002.prelodge.request” | 1. MIG | 1. CMN.ATO.GEN.100002 |

* + - 1. Standard Business Document Body Content

##### IEE.PRELODGE Request XBRL Context

##### Refer to IEE.LODGE request – message.

##### IEE.PRELODGE Request Message Content Table

##### Refer to IEE.LODGE request – message.

* + 1. IEE.PRELODGE Response - Message
       1. Discoverable Taxonomy Set References

Refer to IEE.LODGE response – message.

* + - 1. Standard Business Document Header Content

##### The WIG provides the specification of the SBDH. The following table specifies the message specific data element values or any variations to the WIG for the IEE.

| **Attribute Name** | **Optionality** | **Rules/Values** |
| --- | --- | --- |
| sbdm:Message.Type.Text | Mandatory | Value will be “iee.0002.prelodge.response” |

* + - 1. Standard Business Document Body Content

##### The following describes the facts and context required to be supplied within the XBRL instance document populated into the SBDB element BusinessDocument.Instance.Text

##### IEE.PRELODGE Response XBRL Context

Refer to IEE.LODGE response - message

* 1. IEE.LODGE SPECIFICATIONS

|  |  |
| --- | --- |
| **Interaction Name** | iee.lodge |
| **Description** | The purpose of this transaction is to allow a client to lodge an Interposed Entity Election or Revocation, where it is sent in as a stand-alone form, for processing. |
| **Stakeholders** | Reporting party, Intermediary, ATO |
| **Pre-conditions** | As per the Pre-requisites in Section 5.1 |
| **Post-conditions** | For a successful lodgement the ATO will return:   * A message event item with receipt details informing successful lodgement.   For an unsuccessful lodgement validation the ATO will return:   * A message event item(s) containing a list of errors.   The Reporting party/Intermediary is then able to correct their Schedule for re-submission or to confirm their lodgement has been accepted for processing. |
| **Initiating party** | Reporting party, Intermediary |
| **Channel** | SBR |
| **Core Service Map** | Lodge |

* + 1. IEE.LODGE Request - Message
       1. Discoverable Taxonomy Set References

|  |  |
| --- | --- |
| **Schema** | iee.0002.lodge.request.02.00.report.xsd  iee.0002.private.02.00.module.xsd |
| **Linkbases** | iee.0002.lodge.request.02.00.defLink.xml  iee.0002.private.02.00.defLink.xml |
| iee.0002.lodge.request.02.00.labLinkInfoCls.xml  iee.0002.private.02.00.labLinkInfoCls.xml |
| iee.0002.lodge.request.02.00.presLink.xml |
| iee.0002.lodge.request.02.00.refLink.xml  iee.0002.private.02.00.refLink.xml |
| **Example Instance** | Refer to ATO-CTR Taxonomy Level Conformance Test Suite |
| **Schematron** | To be advised |

* + - 1. Standard Business Document Header Content

##### The WIG provides the specification of the SBDH. The following table specifies the message specific data element values or any variations to the WIG applicable to the IEE.

| **Attribute Name** | **Instructions / Rules** | **Rule implementation** | **SBR message code** |
| --- | --- | --- | --- |
| sbdm:Message.Type.Text | Value must be “iee.0002.lodge.request” | 1. MIG | 1. CMN.ATO.GEN.100002 |
| sbdm:BusinessDocument.BusinessGenerated Identifier.Text | Must be set to a unique identifier for the document being lodged. | 1. MIG | N/A |

* + - 1. Standard Business Document Body Content

The following sections describe the facts and context required in the instance document populated into the SBDB element BusinessDocument.Instance.Text.

* + - 1. IEE.LODGE Request XBRL Context Instances

| **Context spec** | **Hypercube Name(s)** | **Associated Segment DImensions** | **Instructions / Rules** | **Rule Implementation** | **SBR Message Code** |
| --- | --- | --- | --- | --- | --- |
| RP | ReportingParty  ReportingPartyOrIntermediary | RprtPyType.xx.xx:ReportPartyTypeDimension | Refer to Section 4 XBRL CONTEXT | | |
| INT | Intermediary  ReportingPartyOrIntermediary | RprtPyType.xx.xx:ReportPartyTypeDimension | Refer to Section 4 XBRL CONTEXT | | |

* + - 1. IEE.LODGE Request Message Content Table

The following tables contain the facts required in the instance document. To improve readability, aliases are used in the rules. These appear as a word-number combination enclosed between square brackets, such as ‘[IEE156]’. Validation rule aliases are fully defined in Appendix C.

Rules that apply to all instances of Address, Organisation Name and Person Name tuples are defined separately as ‘common rules’ in Appendix E.

The key differences between the stand-alone IEE and the IEE schedule are described in Appendix D.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Context - INT** | |  |  |  |
| **Seq. No** | **XBRL Fact** | **Instructions / Rules** | **Rule Imp** | **SBR Msg Code** |
| 1 | pyid.xx.xx:Identifiers.TaxAgentNumber.Identifier | 1. IF ((pyid.xx.xx:Identifiers.TaxAgentNumber.Identifier) <> NULLORBLANK) AND (TANALGORITHM (Tax Agent number) = FALSE) RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.GEN.410009 | 1. CMN.ATO.GEN.410009 |
| **2** | **prsnunstrcnm1.xx.xx:PersonUnstructuredName (Tuple:0..1)** |  |  |  |
| 2.1 | pyde.xx.xx:PersonUnstructuredName.Usage.Code | 1. IF ([IEE186] <> NULLORBLANK) AND [IEE185] = NULLORBLANK)  RETURN VALIDATION MESSAGE ENDIF 2. WHERE IN TUPLE (xbrli\prsnstrcnm1.xx.xx:PersonUnstructuredName) IF INT:pyde.xx.xx:PersonUnstructuredName.Usage.Code <> NULLORBLANK AND  INT:pyde.xx.xx:PersonUnstructuredName.Usage.Code <> "Contact"  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410179 2. Schematron ID = VR.ATO.IEE.410263 | 1. CMN.ATO.IEE.410179 2. CMN.ATO.IEE.410263 |
| 2.2 | pyde.xx.xx:PersonUnstructuredName.FullName.Text | 1. IF LENGTH(pyde.xx.xx:PersonUnstructuredName.FullName.Text) > 200  RETURN VALIDATION MESSAGE ENDIF 2. IF pyde.xx.xx:PersonUnstructuredName.FullName.Text <> NULLORBLANK AND pyde.xx.xx:PersonUnstructuredName.FullName.Text <> SET("a-z", "A-Z", "-", " ")  RETURN VALIDATION MESSAGE ENDIF 3. IF (pyde.xx.xx:Person UnstructuredName.FullName.Text CONTAINS " - ")  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.GEN.430252 2. Schematron ID = VR.ATO.GEN.430253 3. Schematron ID = VR.ATO.IEE.410221 | 1. CMN.ATO.GEN.430252 2. CMN.ATO.GEN.430253 3. CMN.ATO.GEN.000423 |
| **3** | **declaration2.xx.xx:Declaration (Tuple: 0..1)** |  |  |  |
| 3.1 | pyin.xx.xx:Declaration.StatementType.Code | 1. IF(pyin.xx.xx:Declaration.Statement Type.Code = NULLORBLANK) OR (pyin.xx.xx:Declaration.Statement Accepted.Indicator = NULL) OR (pyin.xx.xx:Declaration.Signature.Date = NULL)  RETURN VALIDATION MESSAGE  ENDIF 2. IF [IEE159] <> NULLORBLANK AND [IEE159] <> "TrueAndCorrect"  RETURN VALIDATION MESSAGE END IF | 1. Schematron ID = VR.ATO.IEE.410266 2. Schematron ID = VR.ATO.IEE.410268 | 1. CMN.ATO.IEE.410266 2. CMN.ATO.IEE.410265 |
| 3.2 | pyin.xx.xx:Declaration.StatementAccepted.Indicator | 1. WHERE PARENT RETURN EXISTS IF INT <> NULLORBLANK AND COUNT [IEE159] > 0  RETURN VALIDATION MESSAGE ENDIF 2. IF [IEE179] = FALSE  RETURN VALIDATION MESSAGE END IF | 1. Schematron ID = VR.ATO.IEE.410262 2. Schematron ID = VR.ATO.IEE.410267 | 1. CMN.ATO.IEE.410262 2. CMN.ATO.IEE.410264 |
| 3.3 | pyin.xx.xx:Declaration.Statement.Text | N/A | N/A | N/A |
| 3.4 | pyin.xx.xx:Declaration.Signature.Date | 1. IF pyde.xx.xx:Declaration.Signature.Date <> NULL AND pyde.xx.xx:Declaration.Signature.Date > DATE(TODAY)  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.GEN.430255 | 1. CMN.ATO.GEN.430255 |
| 3.5 | pyin.xx.xx:Declaration.SignatoryIdentifier.Text | N/A | N/A | N/A |
| **3.6** | **prsnunstrcnm1.xx.xx:PersonUnstructuredName (Tuple:0..1)** |  |  |  |
| 3.6.1 | pyde.xx.xx:PersonUnstructuredName.Usage.Code | 1. WHERE IN TUPLE (personunstructuredname1.xx.xx:PersonUnstructuredName) IN TUPLE(declaration2.xx.xx:Declaration) IF pyde.xx.xx:PersonUnstructuredName.Usage.Code <> "DeclarationSignatory"  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.GEN.436279 | 1. CMN.ATO.GEN.436279 |
| 3.6.2 | pyde.xx.xx:PersonUnstructuredName.FullName.Text | 1. IF LENGTH(pyde.xx.xx:PersonUnstructuredName.FullName.Text) > 200  RETURN VALIDATION MESSAGE ENDIF 2. IF pyde.xx.xx:PersonUnstructuredName.FullName.Text <> NULLORBLANK AND pyde.xx.xx:PersonUnstructuredName.FullName.Text <> SET("a-z", "A-Z", "-", " ")  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.GEN.430252 2. Schematron ID = VR.ATO.GEN.430253 | 1. CMN.ATO.GEN.430252 2. CMN.ATO.GEN.430253 |
| **4** | **electroniccontacttelephone1.xx.xx:ElectronicContactTelephone (Tuple:0..1)** |  |  |  |
| 4.1 | pyde.xx.xx:ElectronicContact.Telephone.Usage.Code | N/A | N/A | N/A |
| 4.2 | pyde.xx.xx:ElectronicContact.Telephone.ServiceLine.Code | N/A | N/A | N/A |
| 4.3 | pyde.xx.xx:ElectronicContact.Telephone.Minimal.Number | 1. WHERE IN TUPLE(electroniccontacttelephone1.xx.xx:ElectronicContactTelephone) IF pyde.xx.xx:ElectronicContact.Telephone.Minimal.Number = NULL OR pyde.xx.xx:ElectronicContact.Telephone.Usage.Code <> "03" OR pyde.xx.xx:Electronic Contact.Telephone.ServiceLine.Code <> SET ("01", "02")  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.GEN.432016 | 1. CMN.ATO.GEN.432016 |
| 4.4 | pyde.xx.xx:ElectronicContact.Telephone.Area.Code | N/A | N/A | N/A |
|  |  |  |  |  |
| **Context - RP** | |  |  |  |
| **Seq. No** | **XBRL Fact** | **Instructions / Rules** | **Rule Imp** | **SBR Msg Code** |
| 1 | pyid.xx.xx:Identifiers.AustralianBusinessNumber.Identifier | 1. IF (RP:pyid.xx.xx:Identifiers.AustralianBusinessNumber.Identifier <> NULLORBLANK) AND (PARENT RETURN:RP:pyid.xx.xx:Identifiers.AustralianBusinessNumber.Identifier <> NULLORBLANK) AND (RP:pyid.xx.xx:Identifiers.AustralianBusinessNumber.Identifier <> PARENT RETURN:RP:pyid.xx.xx:Identifiers.AustralianBusinessNumber.Identifier) RETURN VALIDATION MESSAGE ENDIF 2. IF (pyid.xx.xx:IdentifiersAustralianBusinessNumberIdentifer <> NULLORBLANK) and (ABNALGORITHM(pyid.xx.xx:IdentifiersAustralianBusinessNumber.Identifer) = FALSE) RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.GEN.402010 2. Schematron ID = VR.ATO.GEN.428040 | 1. CMN.ATO.GEN.402010 2. CMN.ATO.GEN.000477 |
| 2 | pyid.xx.xx:Identifiers.AustralianCompanyNumber.Identifier | 1. IF (pyid.xx.xx:Identifiers.AustralianCompanyNumber.Identifier <> NULLORBLANK) AND (ACNALGORITHM(pyid.xx.xx:Identifiers.AustralianCompanyNumber.Identifier) = FALSE)  RETURN VALIDATION MESSAGE ENDIF 2. IF ([IEE100] <> NULLORBLANK) AND ([IEE101] <> NULLORBLANK)  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.GEN.428021 2. Schematron ID = VR.ATO.IEE.410227 | 1. CMN.ATO.GEN.428021 2. CMN.ATO.IEE.410222 |
| 3 | pyid.xx.xx:Identifiers.AustralianRegisteredBodyNumber.Identifier | 1. IF (pyid.xx.xx:Identifiers.AustralianRegisteredBodyNumber.Identifier <> NULLORBLANK) AND (ARBNALGORITHM (pyid.xx.xx:Identifiers.AustralianRegisteredBodyNumber.Identifier) = FALSE)  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.GEN.410105 | 1. CMN.ATO.GEN.410105 |
| 4 | pyde.xx.xx:OrganisationDetails.OrganisationBranch.Code | N/A | N/A | N/A |
| 5 | pyde.xx.xx:Party.Type.Code | 1. IF ([IEE16] <> NULLORBLANK) AND ([IEE16] <> SET("051","161","212","217"))  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410019 | 1. CMN.ATO.IEE.410019 |
| **6** | **orgname2.xx.xx:OrganisationNameDetails (Tuple: 0..2)** |  |  |  |
| 6.1 | pyde.xx.xx:OrganisationNameDetails.OrganisationalNameType.Code | 1. IEE standard organisation rules apply to this tuple 2. IF (pyde.xx.xx:OrganisationNameDetails.OrganisationalNameType.Code <> NULLORBLANK) AND (pyde.xx.xx:OrganisationNameDetails.OrganisationalNameType.Code <> "MN")  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = IEE Ruleset:orgname2 2. Schematron ID = VR.ATO.GEN.410049 | 2. CMN.ATO.GEN.410049 |
| 6.2 | pyde.xx.xx:OrganisationNameDetails.Currency.Code | N/A | N/A | N/A |
| 6.3 | pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text | 1. WHERE PARENT RETURN DOES NOT EXIST IF ([IEE4] = NULLORBLANK)  RETURN VALIDATION MESSAGE ENDIF 2. IF COUNT([IEE4]) >1 OR COUNT([IEE10]) >1  RETURN VALIDATION MESSAGE ENDIF 3. WHERE PARENT RETURN EXISTS IF ([IEE10] <> NULLORBLANK) AND ([IEE10] <> (PARENT RETURN:RP:pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text IN TUPLE (xbrli\orgname2.xx.xx:OrganisationNameDetails) WHERE ((TUPLE EXPLICIT pyde.xx.xx:OrganisationNameDetails.OrganisationalNameType.Code = "MN") AND (TUPLE EXPLICIT pyde.xx.xx:OrganisationNameDetails.Currency.Code = "P")))   RETURN VALIDATION MESSAGE ENDIF 4. WHERE PARENT RETURN EXISTS IF ([IEE4] <> NULLORBLANK) AND ([IEE4] <> (PARENT RETURN:RP:pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text IN TUPLE (xbrli\orgname2.xx.xx:OrganisationNameDetails) WHERE ((TUPLE EXPLICIT pyde.xx.xx:OrganisationNameDetails.OrganisationalNameType.Code = "MN") AND (TUPLE EXPLICIT pyde.xx.xx:OrganisationNameDetails.Currency.Code = "C")))   RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410135 2. Schematron ID = VR.ATO.IEE.410150 3. Schematron ID = VR.ATO.IEE.410225 4. Schematron ID = VR.ATO.IEE.410226 | 1. CMN.ATO.IEE.410135 2. CMN.ATO.IEE.410150 3. CMN.ATO.GEN.402063 4. CMN.ATO.GEN.402063 |
| **7** | **address2.xx.xx:AddressDetails (Tuple:0..2)** |  |  |  |
| 7.1 | pyde.xx.xx:AddressDetails.OverseasAddress.Indicator | 1. IEE standard address rules apply to this tuple | 1. Schematron ID = IEE Ruleset:address2 | N/A |
| 7.2 | pyde.xx.xx:AddressDetails.Usage.Code | 1. NOT IN TUPLE(iee.0002.lodge.request.xx.xx:SpecifiedIndividual)  IF (pyde.xx.xx:AddressDetails.Usage.Code <> NULLORBLANK) AND (pyde.xx.xx:AddressDetails.Usage.Code <>"POS")  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410106 | 1. CMN.ATO.IEE.410106 |
| 7.3 | pyde.xx.xx:AddressDetails.Currency.Code | N/A | N/A | N/A |
| 7.4 | pyde.xx.xx:AddressDetails.Line1.Text | 1. IF COUNT([IEE5]) >1 OR COUNT([IEE11]) >1  RETURN VALIDATION MESSAGE END IF 2. WHERE PARENT RETURN DOES NOT EXIST IF ([IEE5] = NULLORBLANK)  RETURN VALIDATION MESSAGE ENDIF  3. WHERE PARENT RETURN EXISTS IF ([IEE5] <> NULLORBLANK) AND ([IEE5] <> (PARENT RETURN:RP:pyde.xx.xx:AddressDetails.Line1.Text WHERE ((TUPLE ELEMENT EXPLICIT pyde.xx.xx:AddressDetails.Usage.Code = "POS") AND (TUPLE ELEMENT EXPLICIT pyde.xx.xx:AddressDetails.Currency.Code = "C")) IN TUPLE(xbrli\address2.xx.xx:AddressDetails)  RETURN VALIDATION MESSAGE ENDIF 4. WHERE PARENT RETURN EXISTS IF ([IEE11] <> NULLORBLANK) AND ([IEE11] <> (PARENT RETURN:RP:pyde.xx.xx:AddressDetails.Line1.Text WHERE ((TUPLE ELEMENT EXPLICIT pyde.xx.xx:AddressDetails.Usage.Code = "POS") AND (TUPLE ELEMENT EXPLICIT pyde.xx.xx:AddressDetails.Currency.Code = "P")) IN TUPLE(xbrli\address2.xx.xx:AddressDetails)  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410120 2. Schematron ID = VR.ATO.IEE.410136 3. Schematron ID = VR.ATO.IEE.410223 4. Schematron ID = VR.ATO.IEE.410224 | 1. CMN.ATO.IEE.410120 2. CMN.ATO.IEE.410136 3. CMN.ATO.IEE.410223 4. CMN.ATO.IEE.410223 |
| 7.5 | pyde.xx.xx:AddressDetails.Line2.Text | N/A | N/A | N/A |
| 7.6 | pyde.xx.xx:AddressDetails.Line3.Text | N/A | N/A | N/A |
| 7.7 | pyde.xx.xx:AddressDetails.Line4.Text | N/A | N/A | N/A |
| 7.8 | pyde.xx.xx:AddressDetails.LocalityName.Text | 1. WHERE PARENT RETURN DOES NOT EXIST IF ([IEE6] = NULLORBLANK)  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410137 | 1. CMN.ATO.IEE.410137 |
| 7.9 | pyde.xx.xx:AddressDetails.Postcode.Text | N/A | N/A | N/A |
| 7.10 | pyde.xx.xx:AddressDetails.StateOrTerritory.Code | N/A | N/A | N/A |
| 7.11 | pyde.xx.xx:AddressDetails.CountryName.Text | N/A | N/A | N/A |
| 7.12 | pyde.xx.xx:AddressDetails.Country.Code | N/A | N/A | N/A |
| 8 | pyin.xx.xx:Lodgment.PeriodYear.Year | 1. IF (RP:pyin.xx.xx.Lodgment.PeriodYear.Year = NULLORBLANK)  RETURN VALIDATION MESSAGE ENDIF 2. IF (RP:pyin.xx.xx.Lodgment.PeriodYear.Year <> NULLORBLANK) AND (RP:pyin.xx.xx:Lodgment.PeriodYear.Year < "2011")  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410230 2. Schematron ID = VR.ATO.IEE.410231 | 1. CMN.ATO.GEN.001001 2. CMN.ATO.IEE.410231 |
| 9 | rvctc3.xx.xx:Elections.InterposedEntityElectionOrRevocation.Code | 1. IF ([IEE116] = "E") AND (([IEE16] = NULLORBLANK) OR ([IEE45] = NULLORBLANK) OR ([IEE46] = NULL) OR ([IEE52] = NULLORBLANK) OR ([IEE53] = NULLORBLANK) OR ([IEE54] = NULLORBLANK) OR ([IEE58] = NULLORBLANK) OR ([IEE59] = NULL) OR ([IEE64] = NULL) OR ([IEE66] = NULLORBLANK) OR ([IEE70] = NULLORBLANK) OR ([IEE71] = NULLORBLANK) OR ([IEE119] = NULLORBLANK) OR ([IEE170] = NULLORBLANK))  RETURN VALIDATION MESSAGE ENDIF  2. IF ([IEE116] <> NULLORBLANK) AND ([IEE116] <> SET("E","R"))  RETURN VALIDATION MESSAGE ENDIF  3. WHERE PARENT RETURN EXISTS IF ([IEE116] = "R") AND (([IEE52] = NULLORBLANK) OR ([IEE53] = NULLORBLANK) OR ([IEE54] = NULLORBLANK) OR ([IEE119] = NULLORBLANK) OR ([IEE170] = NULLORBLANK) OR ([IEE82] = NULLORBLANK))  RETURN VALIDATION MESSAGE ENDIF  4. IF ([IEE116] = NULLORBLANK)  RETURN VALIDATION MESSAGE ENDIF  5. WHERE PARENT RETURN DOES NOT EXIST IF [IEE116] = "R"  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410006 2. Schematron ID = VR.ATO.IEE.410007 3. Schematron ID = VR.ATO.IEE.410071 4. Schematron ID = VR.ATO.IEE.410138 5. Schematron ID = VR.ATO.IEE.410197 | 1. CMN.ATO.IEE.410006 2. CMN.ATO.IEE.410007 3. CMN.ATO.IEE.410071 4. CMN.ATO.IEE.410138 5. CMN.ATO.IEE.410197 |
| 10 | pyde.xx.xx:Residency.CentralManagementAndControlOutsideAustralia.Indicator | 1. IF ([IEE116] = "E") AND ([IEE16] = SET("217","212")) AND ([IEE17] = NULL)  RETURN VALIDATION MESSAGE ENDIF 2. IF ([IEE116] = "E") AND ([IEE16] = SET("051","161")) AND ([IEE17] <> NULL)   RETURN VALIDATION MESSAGE ENDIF 3. IF ([IEE17] = TRUE) AND ([IEE18] = NULL) AND ([IEE19] = NULL) AND ([IEE20] = NULL)   RETURN VALIDATION MESSAGE ENDIF 4. IF ([IEE17] = FALSE) AND (([IEE18] <> NULL) OR ([IEE19] <> NULL) OR ([IEE20] <> NULL))   RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410004 2. Schematron ID = VR.ATO.IEE.410015 3. Schematron ID = VR.ATO.IEE.410075 4. Schematron ID = VR.ATO.IEE.410083 | 1. CMN.ATO.IEE.410004 2. CMN.ATO.IEE.410015 3. CMN.ATO.IEE.410075 4. CMN.ATO.IEE.410083 |
| 11 | pyde.xx.xx:Residency.CentralManagementAndControlOutsideAustraliaFullPeriod.Indicator | 1. IF ([IEE18] = "TRUE" AND ([IEE19] <> NULL) AND ([IEE20] <> NULL)   RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410088 | 1. CMN.ATO.IEE.410088 |
| 12 | pyde.xx.xx:Residency.TaxPurposesOrganisationStatus.Code | 1. IF ([IEE16] = "051") AND ([IEE21] = NULLORBLANK)  RETURN VALIDATION MESSAGE ENDIF  2. IF ([IEE21] <> NULL) AND ([IEE116] = "E") AND ([IEE16] = SET("161","217","212"))   RETURN VALIDATION MESSAGE ENDIF 3. IF ([IEE21] = SET("02","03")) AND ([IEE22] = NULL AND ([IEE23] = NULL) AND ([IEE24] = NULL)  RETURN VALIDATION MESSAGE ENDIF 4. IF ([IEE21] = "01") AND (([IEE22] <> NULL) OR ([IEE23] <> NULL) OR ([IEE24] <> NULL))  RETURN VALIDATION MESSAGE ENDIF 5. IF ([IEE21] <> NULL) AND ([IEE21] <> SET ("01","02","03"))  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410034 2. Schematron ID = VR.ATO.IEE.410035 3. Schematron ID = VR.ATO.IEE.410089 4. Schematron ID = VR.ATO.IEE.410090 5. Schematron ID = VR.ATO.IEE.410232 | 1. CMN.ATO.IEE.410034 2. CMN.ATO.IEE.410035 3. CMN.ATO.IEE.410089 4. CMN.ATO.IEE.410090 5. CMN.ATO.IEE.410232 |
| 13 | pyde.xx.xx:Residency.NonResidentFullPeriod.Indicator | 1. IF ([IEE22] = TRUE) AND ([IEE23] <> NULL) AND ([IEE24] <> NULL)   RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410091 | 1. CMN.ATO.IEE.410091 |
| 14 | rvctc3.xx.xx:Elections.InterposedEntityElectionStatus.Year | 1. IF ([IEE45] <> NULLORBLANK) AND ([IEE45] < "2011")  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410079 | 1. CMN.ATO.IEE.410233 |
| 15 | rvctc3.xx.xx:Elections.DateSpecified.Date | 1. IF ([IEE49] < [IEE46])   RETURN VALIDATION MESSAGE ENDIF 2. IF ([IEE46] <> NULL) AND ([IEE60] <> NULL) AND ([IEE46] < [IEE60])   RETURN VALIDATION MESSAGE ENDIF  3. IF ([IEE45] <> NULLORBLANK) AND ([IEE46] <> NULL) AND ((([IEE47] = NULL) AND ([IEE46] < ("0107"&([IEE45] - 1)))) OR (([IEE48] = NULL) AND ([IEE46] > ("3006"&([IEE45])))) OR (([IEE47] <> NULL) AND ([IEE46] < [IEE47])) OR (([IEE48] <> NULL) AND ([IEE46] > [IEE48])))   RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410059 2. Schematron ID = VR.ATO.IEE.410073 3. Schematron ID = VR.ATO.IEE.410074 | 1. CMN.ATO.IEE.410059 2. CMN.ATO.IEE.410073 3. CMN.ATO.IEE.410074 |
| 16 | rvctc3.xx.xx:Elections.Commencement.Date | 1. WHERE PARENT RETURN EXISTS IF ([IEE49] <> NULL) AND ([IEE45] <> PARENT RETURN:Report.TargetFinancial.Year)   RETURN VALIDATION MESSAGE ENDIF 2. WHERE PARENT RETURN DOES NOT EXIST IF ([IEE49] <> NULL) AND ([IEE45] <> [IEE1])   RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410051 2. Schematron ID = VR.ATO.IEE.410199 | 1. CMN.ATO.IEE.410235 2. CMN.ATO.IEE.410051 |
| 17 | rvctc3.xx.xx:Elections.Revocation.Year | 1. WHERE PARENT RETURN EXISTS IF ([IEE82] <> NULLORBLANK) AND ([IEE116] = "E")  RETURN VALIDATION MESSAGE ENDIF  2. WHERE PARENT RETURN EXISTS IF ([IEE116] = "R") AND ([IEE82] <> PARENT RETURN:Report.TargetFinancial.Year)   RETURN VALIDATION MESSAGE ENDIF 3. WHERE PARENT RETURN EXISTS IF (([IEE58] <> NULLORBLANK) OR ([IEE59] <> NULL) OR ([IEE60] <> NULL) OR ([IEE61] <> NULL) OR ([IEE62] <> NULLORBLANK) OR ([IEE69] <> NULLORBLANK) OR ([IEE64] <> NULL) OR ([IEE65] <> NULLORBLANK) OR ([IEE66] <> NULLORBLANK) OR ([IEE67] <> NULLORBLANK) OR ([IEE68] <> NULLORBLANK) OR ([IEE69] <> NULLORBLANK) OR ([IEE70] <> NULLORBLANK) OR ([IEE71] <> NULLORBLANK) OR ([IEE72] <> NULLORBLANK) OR ([IEE73] <> NULLORBLANK) AND ([IEE82] <> NULLORBLANK)  RETURN VALIDATION MESSAGE ENDIF 4. WHERE PARENT RETURN EXISTS IF ([IEE82] <> NULLORBLANK) AND ([IEE82] < "2011")  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410069 2. Schematron ID = VR.ATO.IEE.410070 3. Schematron ID = VR.ATO.IEE.410076 4. Schematron ID = VR.ATO.IEE.410084 | 1. CMN.ATO.IEE.410069 2. CMN.ATO.IEE.410070 3. CMN.ATO.IEE.410076 4. CMN.ATO.IEE.410238 |
| 18 | pyin.xx.xx:Report.CompletionHours.Number | 1. IF (pyin.xx.xx:Report.CompletionHours.Number) <> SET(1-999)  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.GEN.410082 | 1. CMN.ATO.GEN.410082 |
| **19** | **period1.xx.xx:PeriodDetails (Tuple: 0..9)** |  |  |  |
| 19.1 | pyin.xx.xx:Period.Type.Code | 1. IF (pyin.xx.xx:Period.Type.Code <> NULLORBLANK) AND (pyin.xx.xx:Period.Type.Code <> SET("ControlOutsideAustraliaPartPeriod","NonResidentPartPeriod","SubstitutedAccountingPeriod")  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.GEN.410116 | 1. CMN.ATO.GEN.410116 |
| 19.2 | pyin.xx.xx:Period.Start.Date | 1. IF ([IEE19] < [IEE46]) OR ([IEE19] < [IEE49])   RETURN VALIDATION MESSAGE ENDIF 2. IF (([IEE23] < [IEE46]) OR ([IEE23] < [IEE49]))   RETURN VALIDATION MESSAGE ENDIF  3. IN TUPLE(period1.xx.xx:PeriodDetails) IF pyin.xx.xx:PeriodStart.Date >= pyin.xx.xx:PeriodEnd.Date  RETURN VALIDATION MESSAGE END IF  4. IN TUPLE(period1.xx.xx:PeriodDetails) IF (pyin.xx.xx:PeriodStart.Date = NULLORBLANK) OR (pyin.xx.xx:PeriodEnd.Date = NULLORBLANK) OR (pyin.xx.xx:Period.Type.Code = NULLORBLANK)  RETURN VALIDATION MESSAGE END IF  5. IF COUNT([IEE19]) > 4  RETURN VALIDATION MESSAGE ENDIF 6. IF COUNT([IEE23]) > 4  RETURN VALIDATION MESSAGE ENDIF 7. IF COUNT([IEE47]) > 1  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410050 2. Schematron ID = VR.ATO.IEE.410056 3. Schematron ID = VR.ATO.IEE.410068 4. Schematron ID = VR.ATO.IEE.410097 5. Schematron ID = VR.ATO.IEE.410187 6. Schematron ID = VR.ATO.IEE.410188 7. Schematron ID = VR.ATO.IEE.410234 | 1. CMN.ATO.IEE.410050 2. CMN.ATO.IEE.410056 3. CMN.ATO.IEE.410068 4. CMN.ATO.IEE.410097 5. CMN.ATO.IEE.410187 6. CMN.ATO.IEE.410188 7. CMN.ATO.IEE.410234 |
| 19.3 | pyin.xx.xx:Period.End.Date | N/A | N/A | N/A |
| **20** | **iee.0002.lodge.req.xx.xx:TrusteeOrPartner (Tuple: 0..10)** |  |  |  |
| 20.1 | pyid.xx.xx:Identifiers.AustralianCompanyNumber.Identifier | 1. IF (pyid.xx.xx:Identifiers.AustralianCompanyNumber.Identifier <> NULLORBLANK) AND (ACNALGORITHM(pyid.xx.xx:Identifiers.AustralianCompanyNumber.Identifier) = FALSE)  RETURN VALIDATION MESSAGE ENDIF 2. IF ([IEE116] = "E") AND ([IEE16] = "051") AND ([IEE34] <> NULLORBLANK)  RETURN VALIDATION MESSAGE ENDIF 3. IF ([IEE34] <> NULLORBLANK) AND ([IEE95] <> NULLORBLANK)  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.GEN.428021 2. Schematron ID = VR.ATO.IEE.410026 3. Schematron ID = VR.ATO.IEE.410222 | 1. CMN.ATO.GEN.428021 2. CMN.ATO.IEE.410026 3. CMN.ATO.IEE.410222 |
| 20.2 | pyid.xx.xx:Identifiers.AustralianRegisteredBodyNumber.Identifier | 1. IF (pyid.xx.xx:Identifiers.AustralianRegisteredBodyNumber.Identifier <> NULLORBLANK) AND (ARBNALGORITHM (pyid.xx.xx:Identifiers.AustralianRegisteredBodyNumber.Identifier) = FALSE)  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.GEN.410105 | 1. CMN.ATO.GEN.410105 |
| 20.3 | pyid.xx.xx:Identifiers.TaxFileNumber.Identifier | 1. IF (pyid.xx.xx:Identifiers.TaxFileNumber.Identifier <> NULLORBLANK) AND (TFNALGORITHM (pyid.xx.xx:Identifiers.TaxFileNumber.Identifier) = FALSE)  RETURN VALIDATION MESSAGE ENDIF 2. WHERE PARENT RETURN DOES NOT EXIST IF ([IEE16] = SET("217","161","212")) AND ([IEE26] <> TRUE) AND ([IEE25] = NULLORBLANK)  RETURN VALIDATION MESSAGE ENDIF  3. WHERE PARENT RETURN EXISTS IF ([IEE116] = "E") AND ([IEE16] = SET ("217","212","161")) AND ([IEE26] <> TRUE) AND ([IEE25] = NULLORBLANK)   RETURN VALIDATION MESSAGE ENDIF  4. IF COUNT(iee.0002.lodge.req.xx.xx:TrusteeOrPartner) > 10  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.GEN.410031 2. Schematron ID = VR.ATO.IEE.410099 3. Schematron ID = VR.ATO.IEE.410100 4. Schematron ID = VR.ATO.IEE.410186 | 1. CMN.ATO.GEN.410031 2. CMN.ATO.IEE.410099 3. CMN.ATO.IEE.410100 4. CMN.ATO.IEE.410186 |
| 20.4 | pyid.xx.xx:Identifiers.TaxFileNumberInexistent.Indicator | 1. IF ([IEE116] = "E") AND ([IEE16] = "051") AND ([IEE26] <> NULL)  RETURN VALIDATION MESSAGE ENDIF 2. IF ([IEE26] = TRUE) AND [IEE25] <> NULLORBLANK  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410032 2. Schematron ID = VR.ATO.IEE.410085 | 1. CMN.ATO.IEE.410032 2. CMN.ATO.IEE.410085 |
| 20.5 | pyde.xx.xx:Residency.NonResidentFullPeriod.Indicator | 1. IF ([IEE116] = "E") AND ([IEE16] = "051") AND ([IEE42] <> NULL)  RETURN VALIDATION MESSAGE ENDIF 2. IF ([IEE42] = TRUE) AND ([IEE43] <> NULL) AND ([IEE44] <> NULL)   RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410029 2. Schematron ID = VR.ATO.IEE.410094 | 1. CMN.ATO.IEE.410029 2. CMN.ATO.IEE.410094 |
| 20.6 | pyde.xx.xx:Residency.TaxPurposesPersonStatus.Indicator | 1. IF ([IEE116] = "E") AND ([IEE16] = "051") AND ([IEE41] <> NULL)  RETURN VALIDATION MESSAGE ENDIF 2. IF ([IEE41] = TRUE) AND ([IEE42] <> TRUE) AND ([IEE43] = NULL) AND ([IEE44] = NULL)   RETURN VALIDATION MESSAGE ENDIF 3. IF ([IEE41] = FALSE) AND (([IEE42] = TRUE) OR ([IEE43] <> NULL) OR ([IEE44] <> NULL))   RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410028 2. Schematron ID = VR.ATO.IEE.410092 3. Schematron ID = VR.ATO.IEE.410093 | 1. CMN.ATO.IEE.410028 2. CMN.ATO.IEE.410092 3. CMN.ATO.IEE.410093 |
| **20.7** | **orgname2.xx.xx:OrganisationNameDetails (Tuple: 0..1)** |  |  |  |
| 20.7.1 | pyde.xx.xx:OrganisationNameDetails.OrganisationalNameType.Code | 1. IEE standard organisation rules apply to this tuple 2. IF (pyde.xx.xx:OrganisationNameDetails.OrganisationalNameType.Code <> NULLORBLANK) AND (pyde.xx.xx:OrganisationNameDetails.OrganisationalNameType.Code <> "MN")  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = IEE Ruleset:orgname2 2. Schematron ID = VR.ATO.GEN.410049 | 2. CMN.ATO.GEN.410049 |
| 20.7.2 | pyde.xx.xx:OrganisationNameDetails.Currency.Code | 1. IN TUPLE(iee.0002.lodge.request.xx.xx:TrusteeOrPartner) IN TUPLE(orgname2.xx.xx:OrganisationNameDetails) IF (pyde.xx.xx:OrganisationNameDetails.Currency.Code <> NULLORBLANK) AND (pyde.xx.xx:OrganisationNameDetails.Currency.Code <> "C")  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410142 | 1. CMN.ATO.IEE.410142 |
| 20.7.3 | pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text | 1. IF ([IEE116] = "E") AND ([IEE16] = "051") AND ([IEE32] <> NULLORBLANK)  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410027 | 1. CMN.ATO.IEE.410027 |
| **20.8** | **prsnstrcnm3.xx.xx:PersonNameDetails (Tuple: 0..1)** |  |  |  |
| 20.8.1 | pyde.xx.xx:PersonNameDetails.PersonNameType.Code | 1. IEE standard name rules apply to this tuple 2. IN TUPLE(prsnstrcnm3.xx.xx:PersonNameDetails) IF (pyde.xx.xx:PersonNameDetails.PersonNameType.Code <> NULLORBLANK) AND (pyde.xx.xx:PersonNameDetails.PersonNameType.Code <> "LGL")  RETURN VALIDATION MESSAGE END IF | 1. Schematron ID = IEE Ruleset:prsnstrcnm3 2. Schematron ID = VR.ATO.IEE.410145 | 2. CMN.ATO.IEE.410145 |
| 20.8.2 | pyde.xx.xx:PersonNameDetails.Usage.Code | 1. IN TUPLE(iee.0002.lodge.request.xx.xx:TrusteeOrPartner) IN TUPLE(prsnstrcnm3.xx.xx:PersonNameDetails) IF (pyde.xx.xx:PersonNameDetails.Usage.Code <> NULLORBLANK) AND (pyde.xx.xx:PersonNameDetails.Usage.Code <> "Identifier")  RETURN VALIDATION MESSAGE END IF | 1. Schematron ID = VR.ATO.IEE.410239 | 1. CMN.ATO.IEE.410239 |
| 20.8.3 | pyde.xx.xx:PersonNameDetails.Currency.Code | 1. IN TUPLE(prsnstrcnm3.xx.xx:PersonNameDetails) IF (pyde.xx.xx:PersonNameDetails.Currency.Code <> NULLORBLANK) AND (pyde.xx.xx:PersonNameDetails.Currency.Code <> "C")  RETURN VALIDATION MESSAGE END IF | 1. Schematron ID = VR.ATO.IEE.410164 | 1. CMN.ATO.IEE.410164 |
| 20.8.4 | pyde.xx.xx:PersonNameDetails.Title.Text | N/A | N/A | N/A |
| 20.8.5 | pyde.xx.xx:PersonNameDetails.NameSuffix.Text | N/A | N/A | N/A |
| 20.8.6 | pyde.xx.xx:PersonNameDetails.FamilyName.Text | 1. IF ([IEE28] <> NULLORBLANK) AND ([IEE32] <> NULLORBLANK)   RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410042 | 1. CMN.ATO.IEE.410042 |
| 20.8.7 | pyde.xx.xx:PersonNameDetails.GivenName.Text | N/A | N/A | N/A |
| 20.8.8 | pyde.xx.xx:PersonNameDetails.OtherGivenName.Text | N/A | N/A | N/A |
| **20.9** | **address2.xx.xx:AddressDetails (Tuple:0..1)** |  |  |  |
| 20.9.1 | pyde.xx.xx:AddressDetails.OverseasAddress.Indicator | 1. IEE standard address rules apply to this tuple | 1. Schematron ID = IEE Ruleset:address2 | N/A |
| 20.9.2 | pyde.xx.xx:AddressDetails.Usage.Code | 1. NOT IN TUPLE(iee.0002.lodge.request.xx.xx:SpecifiedIndividual)  IF (pyde.xx.xx:AddressDetails.Usage.Code <> NULLORBLANK) AND (pyde.xx.xx:AddressDetails.Usage.Code <>"POS")  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410106 | 1. CMN.ATO.IEE.410106 |
| 20.9.3 | pyde.xx.xx:AddressDetails.Currency.Code | 1. IN TUPLE((iee.0002.lodge.request.xx.xx:TrusteeOrPartner) OR (iee.0002.lodge.request.xx.xx:FamilyTrust) OR (iee.0002.lodge.request.xx.xx:SpecifiedIndividual))  IN TUPLE(address2.xx.xx:AddressDetails)  IF (pyde.xx.xx:AddressDetails.Currency.Code <> NULLORBLANK) AND (pyde.xx.xx:AddressDetails.Currency.Code <> "C")  RETURN VALIDATION MESSAGE  ENDIF | 1. Schematron ID = VR.ATO.IEE.410139 | 1. CMN.ATO.IEE.410139 |
| 20.9.4 | pyde.xx.xx:AddressDetails.Line1.Text | 1. IF ([IEE116] = "E") AND ([IEE16] = "051") AND ([IEE35] <> NULLORBLANK)  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410025 | 1. CMN.ATO.IEE.410025 |
| 20.9.5 | pyde.xx.xx:AddressDetails.Line2.Text | N/A | N/A | N/A |
| 20.9.6 | pyde.xx.xx:AddressDetails.Line3.Text | N/A | N/A | N/A |
| 20.9.7 | pyde.xx.xx:AddressDetails.Line4.Text | N/A | N/A | N/A |
| 20.9.8 | pyde.xx.xx:AddressDetails.LocalityName.Text | N/A | N/A | N/A |
| 20.9.9 | pyde.xx.xx:AddressDetails.Postcode.Text | N/A | N/A | N/A |
| 20.9.10 | pyde.xx.xx:AddressDetails.StateOrTerritory.Code | N/A | N/A | N/A |
| 20.9.11 | pyde.xx.xx:AddressDetails.CountryName.Text | N/A | N/A | N/A |
| 20.9.12 | pyde.xx.xx:AddressDetails.Country.Code | N/A | N/A | N/A |
| **20.10** | **period1.xx.xx:PeriodDetails (Tuple: 0..4)** |  |  |  |
| 20.10.1 | pyin.xx.xx:Period.Type.Code | 1. IF (pyin.xx.xx:Period.Type.Code <> NULLORBLANK) AND (pyin.xx.xx:Period.Type.Code <> SET("ControlOutsideAustraliaPartPeriod","NonResidentPartPeriod","SubstitutedAccountingPeriod")  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.GEN.410116 | 1. CMN.ATO.GEN.410116 |
| 20.10.2 | pyin.xx.xx:Period.Start.Date | 1. IF ([IEE43] < [IEE46]) OR ([IEE43] < [IEE49])   RETURN VALIDATION MESSAGE ENDIF 2. IN TUPLE(period1.xx.xx:PeriodDetails) IF pyin.xx.xx:PeriodStart.Date >= pyin.xx.xx:PeriodEnd.Date  RETURN VALIDATION MESSAGE END IF  3. IN TUPLE(period1.xx.xx:PeriodDetails) IF (pyin.xx.xx:PeriodStart.Date = NULLORBLANK) OR (pyin.xx.xx:PeriodEnd.Date = NULLORBLANK) OR (pyin.xx.xx:Period.Type.Code = NULLORBLANK)  RETURN VALIDATION MESSAGE END IF  4. IF COUNT([IEE43]) > 4  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410057 2. Schematron ID = VR.ATO.IEE.410068 3. Schematron ID = VR.ATO.IEE.410097 4. Schematron ID = VR.ATO.IEE.410189 | 1. CMN.ATO.IEE.410057 2. CMN.ATO.IEE.410068 3. CMN.ATO.IEE.410097 4. CMN.ATO.IEE.410189 |
| 20.10.3 | pyin.xx.xx:Period.End.Date | N/A | N/A | N/A |
| **21** | **iee.0002.lodge.req.xx.xx:FamilyTrust (Tuple: 1..1)** |  |  |  |
| 21.1 | rvctc3.xx.xx:Elections.Commencement.Date | N/A | N/A | N/A |
| 21.2 | rvctc3.xx.xx:Elections.FamilyTrustElectionStatus.Year | 1. IF ([IEE58] <> NULLORBLANK) AND ([IEE61] <> NULL) AND ([IEE60] = NULL) AND ([IEE46] < ("0107"&([IEE58] - 1)))  RETURN VALIDATION MESSAGE ENDIF 2. IF ([IEE58] <> NULLORBLANK) AND ([IEE58] < "1995")  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410058 2. Schematron ID = VR.ATO.IEE.410080 | 1. CMN.ATO.IEE.410058 2. CMN.ATO.IEE.410236 |
| 21.3 | pyid.xx.xx:Identifiers.TaxFileNumber.Identifier | 1. IF (pyid.xx.xx:Identifiers.TaxFileNumber.Identifier <> NULLORBLANK) AND (TFNALGORITHM (pyid.xx.xx:Identifiers.TaxFileNumber.Identifier) = FALSE)  RETURN VALIDATION MESSAGE ENDIF 2. WHERE PARENT RETURN DOES NOT EXIST IF ([IEE50] <> NULLORBLANK) AND ([IEE50] = [IEE3])  RETURN VALIDATION MESSAGE END IF 3. IF ([IEE116] = "E") AND ([IEE50] = NULLORBLANK) AND ([IEE51] = NULL)   RETURN VALIDATION MESSAGE ENDIF 4. WHERE PARENT RETURN EXISTS IF ([IEE116] = "R") AND ([IEE50] = NULLORBLANK)   RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.GEN.410031 2. Schematron ID = VR.ATO.IEE.410041 3. Schematron ID = VR.ATO.IEE.410066 4. Schematron ID = VR.ATO.IEE.410193 | 1. CMN.ATO.GEN.410031 2. CMN.ATO.IEE.410041 3. CMN.ATO.IEE.410066 4. CMN.ATO.IEE.410193 |
| 21.4 | pyid.xx.xx:Identifiers.TaxFileNumberInexistent.Indicator | N/A | N/A | N/A |
| **21.5** | **orgname2.xx.xx:OrganisationNameDetails (Tuple: 0..1)** |  |  |  |
| 21.5.1 | pyde.xx.xx:OrganisationNameDetails.OrganisationalNameType.Code | 1. IEE standard organisation rules apply to this tuple 2. IF (pyde.xx.xx:OrganisationNameDetails.OrganisationalNameType.Code <> NULLORBLANK) AND (pyde.xx.xx:OrganisationNameDetails.OrganisationalNameType.Code <> "MN")  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = IEE Ruleset:orgname2 2. Schematron ID = VR.ATO.GEN.410049 | 2. CMN.ATO.GEN.410049 |
| 21.5.2 | pyde.xx.xx:OrganisationNameDetails.Currency.Code | 1. IN TUPLE(iee.0002.lodge.request.xx.xx:FamilyTrust) IN TUPLE(orgname2.xx.xx:OrganisationNameDetails) IF (pyde.xx.xx:OrganisationNameDetails.Currency.Code <> "C")  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410143 | 1. CMN.ATO.IEE.410143 |
| 21.5.3 | pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text | N/A | N/A | N/A |
| **21.6** | **address2.xx.xx:AddressDetails (Tuple:1..1)** |  |  |  |
| 21.6.1 | pyde.xx.xx:AddressDetails.OverseasAddress.Indicator | 1. IEE standard address rules apply to this tuple | 1. Schematron ID = IEE Ruleset:address2 | N/A |
| 21.6.2 | pyde.xx.xx:AddressDetails.Usage.Code | 1. NOT IN TUPLE(iee.0002.lodge.request.xx.xx:SpecifiedIndividual)  IF (pyde.xx.xx:AddressDetails.Usage.Code <> NULLORBLANK) AND (pyde.xx.xx:AddressDetails.Usage.Code <>"POS")  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410106 | 1. CMN.ATO.IEE.410106 |
| 21.6.3 | pyde.xx.xx:AddressDetails.Currency.Code | 1. IN TUPLE((iee.0002.lodge.request.xx.xx:TrusteeOrPartner) OR (iee.0002.lodge.request.xx.xx:FamilyTrust) OR (iee.0002.lodge.request.xx.xx:SpecifiedIndividual))  IN TUPLE(address2.xx.xx:AddressDetails)  IF (pyde.xx.xx:AddressDetails.Currency.Code <> NULLORBLANK) AND (pyde.xx.xx:AddressDetails.Currency.Code <> "C")  RETURN VALIDATION MESSAGE  ENDIF | 1. Schematron ID = VR.ATO.IEE.410139 | 1. CMN.ATO.IEE.410139 |
| 21.6.4 | pyde.xx.xx:AddressDetails.Line1.Text | N/A | N/A | N/A |
| 21.6.5 | pyde.xx.xx:AddressDetails.Line2.Text | N/A | N/A | N/A |
| 21.6.6 | pyde.xx.xx:AddressDetails.Line3.Text | N/A | N/A | N/A |
| 21.6.7 | pyde.xx.xx:AddressDetails.Line4.Text | N/A | N/A | N/A |
| 21.6.8 | pyde.xx.xx:AddressDetails.LocalityName.Text | N/A | N/A | N/A |
| 21.6.9 | pyde.xx.xx:AddressDetails.Postcode.Text | N/A | N/A | N/A |
| 21.6.10 | pyde.xx.xx:AddressDetails.StateOrTerritory.Code | N/A | N/A | N/A |
| 21.6.11 | pyde.xx.xx:AddressDetails.CountryName.Text | N/A | N/A | N/A |
| 21.6.12 | pyde.xx.xx:AddressDetails.Country.Code | N/A | N/A | N/A |
| **21.7** | **period1.xx.xx:PeriodDetails (Tuple: 0..1)** |  |  |  |
| 21.7.1 | pyin.xx.xx:Period.Type.Code | 1. IF (pyin.xx.xx:Period.Type.Code <> NULLORBLANK) AND (pyin.xx.xx:Period.Type.Code <> SET("ControlOutsideAustraliaPartPeriod","NonResidentPartPeriod","SubstitutedAccountingPeriod")  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.GEN.410116 | 1. CMN.ATO.GEN.410116 |
| 21.7.2 | pyin.xx.xx:Period.Start.Date | 1. IN TUPLE(period1.xx.xx:PeriodDetails) IF pyin.xx.xx:PeriodStart.Date >= pyin.xx.xx:PeriodEnd.Date  RETURN VALIDATION MESSAGE END IF  2. IN TUPLE(period1.xx.xx:PeriodDetails) IF (pyin.xx.xx:PeriodStart.Date = NULLORBLANK) OR (pyin.xx.xx:PeriodEnd.Date = NULLORBLANK) OR (pyin.xx.xx:Period.Type.Code = NULLORBLANK)  RETURN VALIDATION MESSAGE END IF  3. IF COUNT([IEE60]) > 1  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410068 2. Schematron ID = VR.ATO.IEE.410097 3. Schematron ID = VR.ATO.IEE.410237 | 1. CMN.ATO.IEE.410068 2. CMN.ATO.IEE.410097 3. CMN.ATO.IEE.410237 |
| 21.7.3 | pyin.xx.xx:Period.End.Date | N/A | N/A | N/A |
| **22** | **iee.0002.lodge.req.xx.xx:SpecifiedIndividual (Tuple: 0..1)** |  |  |  |
| 22.1 | pyid.xx.xx:Identifiers.TaxFileNumber.Identifier | 1. IF (pyid.xx.xx:Identifiers.TaxFileNumber.Identifier <> NULLORBLANK) AND (TFNALGORITHM (pyid.xx.xx:Identifiers.TaxFileNumber.Identifier) = FALSE)  RETURN VALIDATION MESSAGE ENDIF 2. IF ([IEE66] <> NULLORBLANK) AND ([IEE62] = NULLORBLANK) AND ([IEE63] = NULL)   RETURN VALIDATION MESSAGE ENDIF 3. WHERE PARENT RETURN DOES NOT EXIST IF ([IEE63] <> TRUE) AND ([IEE62] = NULLORBLANK)  RETURN VALIDATION MESSAGE ENDIF 4. IF ([IEE63] <> NULL) AND ([IEE62] <> NULLORBLANK)   RETURN VALIDATION MESSAGE ENDIF 5. WHERE PARENT RETURN EXISTS IF ([IEE116] = "E") AND ([IEE62] = NULLORBLANK) AND ([IEE63] <> TRUE)   RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.GEN.410031 2. Schematron ID = VR.ATO.IEE.410067 3. Schematron ID = VR.ATO.IEE.410086 4. Schematron ID = VR.ATO.IEE.410104 5. Schematron ID = VR.ATO.IEE.410198 | 1. CMN.ATO.GEN.410031 2. CMN.ATO.IEE.410067 3. CMN.ATO.IEE.410086 4. CMN.ATO.IEE.410104 5. CMN.ATO.IEE.410086 |
| 22.2 | pyid.xx.xx:Identifiers.TaxFileNumberInexistent.Indicator | N/A | N/A | N/A |
| 22.3 | pyde.xx.xx:PersonDemographicDetails.Birth.Date | 1. IF ([IEE64] <> NULL) AND ((([IEE59] <> NULL) AND ([IEE60] = NULL) AND ([IEE64] > [IEE59])) OR (([IEE59] = NULL) AND ([IEE60] <> NULL) AND ([IEE64] > [IEE60])))   RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410072 | 1. CMN.ATO.IEE.410072 |
| **22.4** | **prsnstrcnm3.xx.xx:PersonNameDetails (Tuple:1..1)** |  |  |  |
| 22.4.1 | pyde.xx.xx:PersonNameDetails.PersonNameType.Code | 1. IEE standard name rules apply to this tuple 2. IN TUPLE(prsnstrcnm3.xx.xx:PersonNameDetails) IF (pyde.xx.xx:PersonNameDetails.PersonNameType.Code <> NULLORBLANK) AND (pyde.xx.xx:PersonNameDetails.PersonNameType.Code <> "LGL")  RETURN VALIDATION MESSAGE END IF | 1. Schematron ID = IEE Ruleset:prsnstrcnm3 2. Schematron ID = VR.ATO.IEE.410145 | 2. CMN.ATO.IEE.410145 |
| 22.4.2 | pyde.xx.xx:PersonNameDetails.Usage.Code | 1. IN TUPLE(iee.0002.lodge.request.xx.xx:SpecifiedIndividual) IN TUPLE(prsnstrcnm3.xx.xx:PersonNameDetails) IF (pyde.xx.xx:PersonNameDetails.Usage.Code <> NULLORBLANK) AND (pyde.xx.xx:PersonNameDetails.Usage.Code <> "Identifier")  RETURN VALIDATION MESSAGE END IF | 1. Schematron ID = VR.ATO.IEE.410240 | 1. CMN.ATO.IEE.410239 |
| 22.4.3 | pyde.xx.xx:PersonNameDetails.Currency.Code | 1. IN TUPLE(prsnstrcnm3.xx.xx:PersonNameDetails) IF (pyde.xx.xx:PersonNameDetails.Currency.Code <> NULLORBLANK) AND (pyde.xx.xx:PersonNameDetails.Currency.Code <> "C")  RETURN VALIDATION MESSAGE END IF | 1. Schematron ID = VR.ATO.IEE.410164 | 1. CMN.ATO.IEE.410164 |
| 22.4.4 | pyde.xx.xx:PersonNameDetails.Title.Text | N/A | N/A | N/A |
| 22.4.5 | pyde.xx.xx:PersonNameDetails.NameSuffix.Text | N/A | N/A | N/A |
| 22.4.6 | pyde.xx.xx:PersonNameDetails.FamilyName.Text | N/A | N/A | N/A |
| 22.4.7 | pyde.xx.xx:PersonNameDetails.GivenName.Text | N/A | N/A | N/A |
| 22.4.8 | pyde.xx.xx:PersonNameDetails.OtherGivenName.Text | N/A | N/A | N/A |
| **22.5** | **address2.xx.xx:AddressDetails (Tuple:0..1)** |  |  |  |
| 22.5.1 | pyde.xx.xx:AddressDetails.OverseasAddress.Indicator | 1. IEE standard address rules apply to this tuple | 1. Schematron ID = IEE Ruleset:address2 | N/A |
| 22.5.2 | pyde.xx.xx:AddressDetails.Usage.Code | 1. IN TUPLE(iee.0002.lodge.request.xx.xx:SpecifiedIndividual)  IF (pyde.xx.xx:AddressDetails.Usage.Code <> NULLORBLANK) AND (pyde.xx.xx:AddressDetails.Usage.Code <> "RES")  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410110 | 1. CMN.ATO.IEE.410110 |
| 22.5.3 | pyde.xx.xx:AddressDetails.Currency.Code | 1. IN TUPLE((iee.0002.lodge.request.xx.xx:TrusteeOrPartner) OR (iee.0002.lodge.request.xx.xx:FamilyTrust) OR (iee.0002.lodge.request.xx.xx:SpecifiedIndividual))  IN TUPLE(address2.xx.xx:AddressDetails)  IF (pyde.xx.xx:AddressDetails.Currency.Code <> NULLORBLANK) AND (pyde.xx.xx:AddressDetails.Currency.Code <> "C")  RETURN VALIDATION MESSAGE  ENDIF | 1. Schematron ID = VR.ATO.IEE.410139 | 1. CMN.ATO.IEE.410139 |
| 22.5.4 | pyde.xx.xx:AddressDetails.Line1.Text | N/A | N/A | N/A |
| 22.5.5 | pyde.xx.xx:AddressDetails.Line2.Text | N/A | N/A | N/A |
| 22.5.6 | pyde.xx.xx:AddressDetails.Line3.Text | N/A | N/A | N/A |
| 22.5.7 | pyde.xx.xx:AddressDetails.Line4.Text | N/A | N/A | N/A |
| 22.5.8 | pyde.xx.xx:AddressDetails.LocalityName.Text | N/A | N/A | N/A |
| 22.5.9 | pyde.xx.xx:AddressDetails.Postcode.Text | N/A | N/A | N/A |
| 22.5.10 | pyde.xx.xx:AddressDetails.StateOrTerritory.Code | N/A | N/A | N/A |
| 22.5.11 | pyde.xx.xx:AddressDetails.CountryName.Text | N/A | N/A | N/A |
| 22.5.12 | pyde.xx.xx:AddressDetails.Country.Code | N/A | N/A | N/A |
| **23** | **iee.0002.lodge.req.xx.xx:DeclarationOfTrusteeOrPartner (Tuple:0..10)** |  |  |  |
| **23.1** | **prsnstrcnm3.xx.xx:PersonNameDetails (Tuple:0..1)** |  |  |  |
| 23.1.1 | pyde.xx.xx:PersonNameDetails.PersonNameType.Code | 1. IEE standard name rules apply to this tuple | 1. Schematron ID = IEE Ruleset:prsnstrcnm3 | N/A |
| 23.1.2 | pyde.xx.xx:PersonNameDetails.Usage.Code | N/A | N/A | N/A |
| 23.1.3 | pyde.xx.xx:PersonNameDetails.Currency.Code | N/A | N/A | N/A |
| 23.1.4 | pyde.xx.xx:PersonNameDetails.Title.Text | N/A | N/A | N/A |
| 23.1.5 | pyde.xx.xx:PersonNameDetails.NameSuffix.Text | N/A | N/A | N/A |
| 23.1.6 | pyde.xx.xx:PersonNameDetails.FamilyName.Text | 1. WHERE IN TUPLE(iee.0002.lodge.req.xx.xx:DeclarationOfTrusteeOrPartner) IF ([IEE203] <> NULL) AND ([IEE88] <> NULL)  RETURN VALIDATION MESSAGE  ENDIF 2. WHERE IN TUPLE(iee.0002.lodge.req.xx.xx:DeclarationOfTrusteeOrPartner) IF ([IEE203] = NULL) AND ([IEE88] = NULL)  RETURN VALIDATION MESSAGE  ENDIF 3. WHERE IN TUPLE (iee.0002.lodge.req.xx.xx:DeclarationOfTrusteeOrPartner\prsnstrcnm3.xx.xx:PersonNameDetails)  IF (PersonNameDetails.PersonNameType.Code <> "LGL") OR (PersonNameDetails.Usage.Code <> "DeclarationSignatory") OR (PersonNameDetails.Currency.Code <> "C")  RETURN VALIDATION MESSAGE  ENDIF | 1. Schematron ID = VR.ATO.IEE.410054 2. Schematron ID = VR.ATO.IEE.410247 3. Schematron ID = VR.ATO.IEE.410248 | 1. CMN.ATO.IEE.410246 2. CMN.ATO.IEE.410247 3. CMN.ATO.IEE.410248 |
| 23.1.7 | pyde.xx.xx:PersonNameDetails.GivenName.Text | N/A | N/A | N/A |
| 23.1.8 | pyde.xx.xx:PersonNameDetails.OtherGivenName.Text | N/A | N/A | N/A |
| **23.2** | **orgname2.xx.xx:OrganisationNameDetails (Tuple: 0..1)** |  |  |  |
| 23.2.1 | pyde.xx.xx:OrganisationNameDetails.OrganisationalNameType.Code | 1. IEE standard organisation rules apply to this tuple | 1. Schematron ID = IEE Ruleset:orgname2 | N/A |
| 23.2.2 | pyde.xx.xx:OrganisationNameDetails.Currency.Code | N/A | N/A | N/A |
| 23.2.3 | pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text | 1. WHERE IN TUPLE (iee.0002.lodge.req.xx.xx:DeclarationOfTrusteeOrPartner\orgname2.xx.xx:OrganisationNameDetails)  IF (OrganisationNameDetails.OrganisationalNameType.Code <> "MN") OR (OrganisationNameDetails.Currency.Code <> "C")  RETURN VALIDATION MESSAGE  ENDIF | 1. Schematron ID = VR.ATO.IEE.410249 | 1. CMN.ATO.IEE.410249 |
| **23.3** | **declaration2.xx.xx:Declaration (Tuple: 1..1)** |  |  |  |
| 23.3.1 | pyin.xx.xx:Declaration.StatementType.Code | 1. IF COUNT(iee.0002.lodge.req.xx.xx:DeclarationOfTrusteeOrPartner) > 10  RETURN VALIDATION MESSAGE ENDIF 2. IF COUNT(iee.0002.lodge.req.xx.xx:DeclarationOfTrusteeOrPartner) <> COUNT (iee.0002.lodge.request.xx.xx:TrusteeOrPartner)  RETURN VALIDATION MESSAGE ENDIF 3. WHERE IN TUPLE(iee.0002.lodge.req.xx.xx:DeclarationOfTrusteeOrPartner\declaration2.xx.xx:Declaration) IF (pyin.xx.xx:Declaration.StatementType.Code <> NULLORBLANK) AND (pyin.xx.xx:Declaration.StatementType.Code <> SET ("HardCopy", "TrueAndCorrect" )  RETURN VALIDATION MESSAGE ENDIF  4. WHERE IN TUPLE(iee.0002.lodge.req.xx.xx:DeclarationOfTrusteeOrPartner\declaration2.xx.xx:Declaration) IF COUNT (pyin.xx.xx:Declaration.StatementType.Code ="TrueAndCorrect") <> 1  RETURN VALIDATION MESSAGE ENDIF  5. IF ([IEE16] = "051" AND [IEE193] <> NULLORBLANK)   RETURN VALIDATION MESSAGE ENDIF  6. IF(pyin.xx.xx:Declaration.Statement Type.Code = NULLORBLANK) OR (pyin.xx.xx:Declaration.Statement Accepted.Indicator = NULL) OR (pyin.xx.xx:Declaration.Signature.Date = NULL)  RETURN VALIDATION MESSAGE  ENDIF ENDIF | 1. Schematron ID = VR.ATO.IEE.410250 2. Schematron ID = VR.ATO.IEE.410251 3. Schematron ID = VR.ATO.IEE.410254 4. Schematron ID = VR.ATO.IEE.410255 5. Schematron ID = VR.ATO.IEE.410260 6. Schematron ID = VR.ATO.IEE.410266 | 1. CMN.ATO.IEE.410250 2. CMN.ATO.IEE.410251 3. CMN.ATO.IEE.410254 4. CMN.ATO.IEE.410255 5. CMN.ATO.IEE.410260 6. CMN.ATO.IEE.410266 |
| 23.3.2 | pyin.xx.xx:Declaration.StatementAccepted.Indicator | 1. WHERE IN TUPLE(iee.0002.lodge.req.xx.xx:DeclarationOfTrusteeOrPartner\declaration2.xx.xx:Declaration) IF [IEE146] = FALSE  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410252 | 1. CMN.ATO.IEE.410252 |
| 23.3.3 | pyin.xx.xx:Declaration.Statement.Text | N/A | N/A | N/A |
| 23.3.4 | pyin.xx.xx:Declaration.Signature.Date | 1. IF pyde.xx.xx:Declaration.Signature.Date <> NULL AND pyde.xx.xx:Declaration.Signature.Date > DATE(TODAY)  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.GEN.430255 | 1. CMN.ATO.GEN.430255 |
| 23.3.5 | pyin.xx.xx:Declaration.SignatoryIdentifier.Text | N/A | N/A | N/A |
| **23.3.6** | **prsnunstrcnm1.xx.xx:PersonUnstructuredName (Tuple:0..1)** |  |  |  |
| 23.3.6.1 | pyde.xx.xx:PersonUnstructuredName.Usage.Code | 1. WHERE IN TUPLE(iee.0002.lodge.req.xx.xx:DeclarationOfTrusteeOrPartner\declaration2.xx.xx:Declaration) IF COUNT (TUPLE(prsnstrcnm1.xx.xx:PersonUnstructuredName) > 0)  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410258 | 1. CMN.ATO.IEE.410258 |
| 23.3.6.2 | pyde.xx.xx:PersonUnstructuredName.FullName.Text | N/A | N/A | N/A |
| **24** | **prsnstrcnm3.xx.xx:PersonNameDetails (Tuple:0..1)** |  |  |  |
| 24.1 | pyde.xx.xx:PersonNameDetails.PersonNameType.Code | 1. IEE standard name rules apply to this tuple 2. IN TUPLE(prsnstrcnm3.xx.xx:PersonNameDetails) IF (pyde.xx.xx:PersonNameDetails.PersonNameType.Code <> NULLORBLANK) AND (pyde.xx.xx:PersonNameDetails.PersonNameType.Code <> "LGL")  RETURN VALIDATION MESSAGE END IF | 1. Schematron ID = IEE Ruleset:prsnstrcnm3 2. Schematron ID = VR.ATO.IEE.410145 | 2. CMN.ATO.IEE.410145 |
| 24.2 | pyde.xx.xx:PersonNameDetails.Usage.Code | 1. WHERE IN TUPLE(prsnstrcnm3.xx.xx:PersonNameDetails) IF (pyde.xx.xx:PersonNameDetails.Usage.Code <> NULLORBLANK) AND (pyde.xx.xx:PersonNameDetails.Usage.Code <> "DeclarationSignatory")  RETURN VALIDATION MESSAGE END IF | 1. Schematron ID = VR.ATO.IEE.410241 | 1. CMN.ATO.IEE.410125 |
| 24.3 | pyde.xx.xx:PersonNameDetails.Currency.Code | 1. IN TUPLE(prsnstrcnm3.xx.xx:PersonNameDetails) IF (pyde.xx.xx:PersonNameDetails.Currency.Code <> NULLORBLANK) AND (pyde.xx.xx:PersonNameDetails.Currency.Code <> "C")  RETURN VALIDATION MESSAGE END IF | 1. Schematron ID = VR.ATO.IEE.410164 | 1. CMN.ATO.IEE.410164 |
| 24.4 | pyde.xx.xx:PersonNameDetails.Title.Text | N/A | N/A | N/A |
| 24.5 | pyde.xx.xx:PersonNameDetails.NameSuffix.Text | N/A | N/A | N/A |
| 24.6 | pyde.xx.xx:PersonNameDetails.FamilyName.Text | N/A | N/A | N/A |
| 24.7 | pyde.xx.xx:PersonNameDetails.GivenName.Text | N/A | N/A | N/A |
| 24.8 | pyde.xx.xx:PersonNameDetails.OtherGivenName.Text | N/A | N/A | N/A |
| **25** | **declaration2.xx.xx:Declaration (Tuple: 0..1)** |  |  |  |
| 25.1 | pyin.xx.xx:Declaration.StatementType.Code | 1. IF ([IEE16] = SET ("217","212","161") AND [IEE147] <> NULLORBLANK)   RETURN VALIDATION MESSAGE ENDIF  2. WHERE PARENT RETURN DOES NOT EXIST IF INT<> NULLORBLANK AND COUNT (iee.0002.lodge.req.xx.xx:DeclarationOfTrusteeOrPartner) > 0 OR COUNT [IEE147] > 0  RETURN VALIDATION MESSAGE ENDIF  3. IF [IEE147] <> NULLORBLANK AND [IEE147] <> "TrueAndCorrect"  RETURN VALIDATION MESSAGE END IF 4. IF(pyin.xx.xx:Declaration.Statement Type.Code = NULLORBLANK) OR (pyin.xx.xx:Declaration.Statement Accepted.Indicator = NULL) OR (pyin.xx.xx:Declaration.Signature.Date = NULL)  RETURN VALIDATION MESSAGE  ENDIF | 1. Schematron ID = VR.ATO.IEE.410259 2. Schematron ID = VR.ATO.IEE.410261 3. Schematron ID = VR.ATO.IEE.410265 4. Schematron ID = VR.ATO.IEE.410266 | 1. CMN.ATO.IEE.410259 2. CMN.ATO.IEE.410261 3. CMN.ATO.IEE.410265 4. CMN.ATO.IEE.410266 |
| 25.2 | pyin.xx.xx:Declaration.StatementAccepted.Indicator | 1. IF [IEE192] = FALSE  RETURN VALIDATION MESSAGE  END IF | 1. Schematron ID = VR.ATO.IEE.410264 | 1. CMN.ATO.IEE.410264 |
| 25.3 | pyin.xx.xx:Declaration.Statement.Text | N/A | N/A | N/A |
| 25.4 | pyin.xx.xx:Declaration.Signature.Date | 1. IF pyde.xx.xx:Declaration.Signature.Date <> NULL AND pyde.xx.xx:Declaration.Signature.Date > DATE(TODAY)  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.GEN.430255 | 1. CMN.ATO.GEN.430255 |
| 25.5 | pyin.xx.xx:Declaration.SignatoryIdentifier.Text | N/A | N/A | N/A |
| **25.6** | **prsnunstrcnm1.xx.xx:PersonUnstructuredName (Tuple:0..1)** |  |  |  |
| 25.6.1 | pyde.xx.xx:PersonUnstructuredName.Usage.Code | 1.  IF COUNT [IEE147] > 0 and COUNT (TUPLE(prsnstrcnm1.xx.xx:PersonUnstructuredName) > 0)  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.IEE.410243 | 1. CMN.ATO.IEE.410243 |
| 25.6.2 | pyde.xx.xx:PersonUnstructuredName.FullName.Text | N/A | N/A | N/A |

* + 1. IEE.LODGE Response - Message
       1. Discoverable Taxonomy Set References

No XBRL instance will be returned.

* + - 1. Standard Business Document Header Content

##### The WIG provides the specification of the SBDH. The following table specifies the message specific data element values or any variations to the WIG.

| **Attribute Name** | **Instructions/Rules** |
| --- | --- |
| sbdm:Message.Type.Text | Mandatory. Value will be “iee.0002.lodge.response” |
| sbdm:Lodgement.Receipt.Identifier | Will not be provided |
| sbdm:Lodgement.Receipt.Datetime | Optional. Will be provided for successful lodgements |

* + - 1. Standard Business Document Body Content

##### No XBRL instance will be returned.

Appendix A – The message content table explained

This section defines the table structure that **must** be used to define the context, structure, and rules of the data elements contained within the XBRL instance document – referred to as the message content table.

There will be a message content table for each context within the message. The grouping of the data elements in accordance to the context aligns to how the data elements are built into the XBRL taxonomy and this consistent presentation will assist the software developer.

The message content table uses the following rows and columns:

**Context Type (row at top of table)**: This is the name of the XBRL Context Specification or Context Instances which have been defined early in the MIG document.

**Sequence Number:** This is a sequential number used to indicate the expected order of the data elements within the instance document and to describe the structure of tuples. Data elements within a tuple are allocated a multilevel number to indicate the expected structure and order of the data elements contained within the tuple. If there are nested tuples then the sequence number goes to next numbering level. The following is an extract from a message content table which illustrates the sequence numbering and multi-levelling required to cater for tuples and nested tuples.

| **RP (MIG context label)** | | | |
| --- | --- | --- | --- |
| **XBRL fact** | **Instructions / Rules** | **Rule Imp** | **SBR Msg Code** |
| Organisation Details.State Government Agency.Indicator |  |  |  |
| PayrollTaxPaymentMechanism (Tuple 1..1) |  |  |  |
| Payment Mechanism.Payment Reference Number.Identifier |  |  |  |
| Payment Mechanism.Payment Method.Code |  |  |  |
| Payment Mechanism.Customer Reference.Number |  |  |  |
| DirectDebit (Tuple 0..n) |  |  |  |
| Payment Mechanism.Direct Debit Name.Text |  |  |  |
| Payment Mechanism.Direct Debit Account.Identifier |  |  |  |
| Preferences.Tax Payment Direct Debit Authorisation.Indicator |  |  |  |
| PayrollTaxPayment (Tuple 0..1) |  |  |  |
| Payment Record.Client Intended Payment.Date |  |  |  |
| Payroll Tax.Payable Tax Calculated.Amount |  |  |  |
| Payroll Tax.Taxable Calculated.Amount |  |  |  |

**XBRL Fact**: This is the name of the data element to be reported. For example:

Identifiers.AustralianBusinessNumber.Identifier

**Instructions/Rules**: This column describes all the instructions/rules applicable to the data element. Each rule needs to be given a sequential number which links the rule to its implementation and message code. Rules would include information such as optionality, presentation criteria and, if being used, the use of XML attributes such as IsVisible.

**Rule Implementation**: This column informs the software developer how the rules specified in the Instructions/Rules column will be provided.

NOTE: This column is only applicable for request messages and the column will not be present in the table for Response Messages.

There can only be the following options:

* XBRL – Validation provided via the XBRL schemas and link-bases. Typically rules implemented via XBRL do not need to be specified within the MIG. Only those rules that are considered to provide necessary information to the software developer should be defined within the MIG. An example of this is the rules associated with the domain values of a dimension within a context specification.
* Schematron ID – For rules that cannot be implemented using XBRL, some agencies will provide a Schematron implementation of the rule. When Schematron is provided then the unique ID used to identify the rule within the Schematron file must be provided within the MIG. The following is an example of how this should appear in the column: Schematron ID = VICMIG001
* MIG – There will be situations where rules will not be provided to the software developer in a machine readable format and the description of the rule in the MIG is all that will be provided. In this situation the software developer has the choice of either implementing the rule as specified within the MIG, or they rely on the agency to validate the data element (the expectation is that the Agency will always test for this rule)
* Agency – This rule cannot be implemented by the software developer and can only be executed by the agency.

**SBR Message Code**: All messages returned via the SBR channel will contain a code to uniquely identify the condition that has occurred.

NOTE: This column is only applicable for request messages and the column will not be present in the table for Response Messages.

In order to allow codes to be managed in a distributed fashion, codes will take the following format:

**{Jurisdiction}.{Agency}.{Function}.{Id}**

Represented by the regular expression:

**([A-Z0-9])+.([A-Z0-9])+.([A-Z0-9])+.([A-Z0-9])+**

Initially

**Jurisdiction** = SBR | CMN | QLD | NSW | ACT | VIC | SA | WA | NT | TAS

**Agency** = Jurisdiction specific agency code

For CMN (Commonwealth) = ATO, ASIC, APRA, ABS

For SBR = GEN (i.e. SBR wide codes)

For States = OSR of Offices of State Revenue

**Function** = Agency specific functional area or GEN for agency wide codes

For SBR = GEN or FAULT

**Id** = function specific identifier (format may vary across agencies).

Examples are shown below;

SBR.GEN.FAULT.TOOMANYINSTANCES

CMN.ATO.TFN.OK

QLD.OSR.PRL.000001

The above structure recognises and caters for the current situation where agency errors are un-harmonised, and will need to be passed through to client software.

The expectation is that for each rule identified within the message content table to have a corresponding message code. However, depending on the rule implementation, a message code may not be relevant in which case Not Applicable (N/A) should be inserted into the rule’s corresponding message code to make this clear to the software developer. The following table summarises what must be provided in the message code column in relation to the rules implementation choice.

|  |  |
| --- | --- |
| **Rule Implementation** | **Message Code** |
| Schematron | Message Code needs to be provided against corresponding rule. |
| XBRL | Message Code not relevant – place N/A against corresponding rule. |
| MIG | Message Code needs to be provided against corresponding rule. The only exception is if the rule is associated to rendering instruction to the software developer. |
| Agency | Message Code needs to be provided against corresponding rule. |

The expectation is that each agency will populate a message repository with all error, warning and information message that could be returned via the SBR channel. These messages will be allocated an SBR message code using the above mentioned code format. The software developer will then use the SBR message code provided via the MIG and the message repository to obtain the full details associated with the message.

Appendix B – Tax Office Structured English

The validation and mapping transformation rules are expressed in structured English. The following table defines terms and characters used throughout these rules.

| **Structured English term** | **Intended interpretation** | **Examples** |
| --- | --- | --- |
| -  (as in <a> - <b>) | Minus. | <a> - <b> Means value of ‘a’ minus the value of ‘b’ |
| -  (as in SET(0-9) | Specifies a range of numeric or alphabetic values within a ‘SET’ construct. | = SET(0-3) Means is equal to 0, 1, 2 or 3  = SET(a-z) Means is equal to a, b, c, d …(etc)… or z  DOES NOT CONTAIN SET(a-z) Means the field does not include any incidence of a lower case alphabetic character between a and z |
| & | Concatenate. Joins the value of a field to a literal or other field | [TRT1]&"-06-30" Where [TRT1] is 2010, means 2010-06-30 |
| +/- | In numerical calculations, allows for an allowable deviation | <a> <> <b> +/- 1 Means (<a> > <b> + 1) OR (<a> < <b> - 1)  <a> = <b> +/- 1 Means (<a> = <b>) OR (<a> = <b> + 1) OR (<a> = <b> - 1) |
| <> | Not equal to | IF <a> <> <b> Means if the value of ‘a’ is not equal to the value of ‘b’ |
| ABSVALUE | Absolute value. Ignore the signage of a numeric value. | IF <a> > ABSVALUE(<b>) Where the value of <b> is 25, means IF <a> > 25. Where the value of <b> is -20, means IF <a> > 20 |
| ALGORITHM (with <IDtype> prefix) | Defines a test against a standard algorithm – as indicated by the <IDtype> prefix. <IDtype> can be ABN, TFN, TAN, ARBN or ACN | IF TFNALGORITHM(<a>) = FALSE Means if the TFN field <a> fails the TFN algorithm.  IF ABNALGORITHM(<a>) = FALSE Means the ABN field <a> fails the ABN algorithm. |
| ALL OCCURENCES OF | All instances of a given field, where the field is from a repeatable tuple. | IF SUM(ALL OCCURRENCES OF(<a> - <b>)) <> <c> Means if the sum of every instance of <a>, minus the sum of every instance of <b> is not equal to the value of <c> (where <a> and <b> are from a repeatable tuple). |
| AND | Logical AND |  |
| ANY OCCURRENCE OF | Any instances of a given field, where the field is from repeatable tuple. | IF ANY OCCURRENCE OF(<a>) > 10 Means if any instance of <a> is greater than 10 (where <a> is from a repeatable tuple). |
| ANY OTHER OCCURRENCE OF | Used to check if any given value for a particular element is repeated in the same element, where the element is part of a repeatable tuple. | IF <a> = ANY OTHER OCCURRENCE OF <a> Means if the value of <a> from one instance of the tuple, is equal to the value of another instance of the tuple. |
| CONTAINS | A text field includes a given string within the field | IF <a> CONTAINS "UNKNOWN" Means if <a> contains or equals the word ‘unknown’. |
| CONTAINS MORE THAN ONE | A text field contains more than one incidence of a given string with the field | IF pyde.xx.xx:ElectronicContact.ElectronicMail.Address.Text CONTAINS MORE THAN ONE "@"  Means if there is more than one ‘@’ symbol within the email address. |
| CONTEXT | Used to refer to a context instance | WHERE CONTEXT = “RPI.Closing” Means in instances where the context instance is ‘RPI.Closing’. |
| COUNT | Count of the number of occurences of a field or context | IF COUNT(RPI) > 1 Means if the number of occurrences of the RPI context is more than 1.  IF COUNT(SCHEDULE = "IEE") > 50 Means if the number of occurrence of an IEE schedule in the business document is greater than 50. |
| DATE(TODAY) | Compares a date against the current date | IF <a> > DATE(TODAY) Means if <a> is a date in the future. |
| DIMENSION | Test against a specific set of metadata for a particular context | IF (RprtPyType.xx.xx:ReportingPartyTypeDimension = “RprtPyType.xx.xx:Intermediary”) Means if the Reporting Party Type context is ‘Intermediary’. |
| DOES NOT CONTAIN | An element has no instance of the stated value or set of values | DOES NOT CONTAIN SET("a-z", "A-Z", "0-9") Means that the field has no instance of an alphabetical character (excepting special characters), nor a numeric character. |
| DOMAIN | A globally defined set of values | SET(DOMAIN(COUNTRY CODES) Means the complete set of country codes. Each set of domain values is defined in the Standard enumerations section within this document. |
| ENDSWITH | The last characters within a text string field are equal to the stated value | IF <a> ENDSWITH " T/A" Means the condition is true if field <a> contains a value that ends with the text string ‘. T/A’. |
| FOUND | A text string field:   * equals one of the stated variables, or * starts with one of the stated variables followed by a space, or * ends with one the stated variables preceded by a space; or * contains one the stated variables preceded by a space and followed by a space. | IF <a> = FOUND("The trustee","The Exec") Means if <a>:   * equals ‘The trustee’ or ‘The Exec’ (exact match), or * contains ‘ The trustee ’ or ‘ The Exec ’ (a space on each side of the variable), or * starts with ‘The trustee ’ or ‘The Exec ’ (with a space after), or * ends with ‘ The trustee’ or ‘ The Exec’ (with a space before). |
| IN TUPLE | Restricts a test to the value of a field within a particular tuple. (Where the field may exist in more than one tuple).  (See also: ‘WHERE’) | IF <a> IN TUPLE(<b>) Means if the value of <a> within the tuple <b>. (Where <a> may also exist outside tuple <b>). |
| LENGTH | Used to define the contraints on the length of a field.  (See also TEXT). | IF LENGTH(<a>) < 6 Means if the value of <a> does not contain at least 6 characters. |
| MONETARY() | Defines a monetary field pattern where a true response is given when a value passes all conditions.  As in: MONETARY(<a>,<b>,<c>) Where:   * <a> = S or U to indicate if field can be signed or not * <b> = Maximum number of digits (including decimal places) * <c> = Maximum number of decimal places   Notes: For <a> an S indicates a field can be prefixed with a sign, but may be omitted.  Where <a> is a U, the field must not be prefixed with a sign.  The value of <b> does not include a decimal point or sign in the total character limit. | <a> <> MONETARY(U,11,0) Field <a> is not equal to a number between 0 and 99999999999, or includes a character other than 0 to 9.  <a> <> MONETARY(S,11,0) Field <a> is not equal to a number between -99999999999 and 99999999999, or includes a character other than 0 to 9, or ‘+’ or ‘–‘ as the first (left-most) character.  <a> <> MONETARY(U,13,2) Field <a> is not equal to a number between 0.00 and 99999999999.99, or includes a character other than 0 to 9 or a decimal point, or a decimal point is not present as the third character from the right. |
| NOT | Reverses the value of a boolean. Turns TRUE to FALSE and vice versa. |  |
| NULL | Fact is not there, is null with xsI:nil = true or is a null non-textual value. | IF <a> = NULL Means if a (non-textual) value for <a> is blank or if <a> does not exist. |
| NULLORBLANK | Fact is not there, is null with xsI:nil = true or is a null string. (Applied to Text, Code, Description and Identifier facts). | IF <a> = NULLORBLANK Means if a (textual) value for <a> is blank or if <a> does not exist. |
| OR | Logical OR |  |
| PARENT RETURN | Used in rules that apply to schedules to refer to a field on the main form | IF <a> <> PARENT RETURN:<a> Means if the value of <a> on the schedule is not equal to the value of <a> on the main form.  WHERE PARENT RETURN EXISTS Means apply the test if this is a business document containing a schedule as a part of a main form. (Applies only to IEE and FTER whch may be submitted either as a form on its own or as a schedule as part of a form). |
| SCHEDULE | Refers to a tax schedule submitted within the same business document as a main tax return form. | IF COUNT(SCHEDULE = "S25A") = 0 Means if there is no instance of a Schedule 25A included in the business document.  IF COUNT(SCHEDULE = "RSPT") > 50 Means if the number of occurrence of a Rental schedule in the business document is greater than 50. |
| SET | Definition of an explicit set of values where if one value meets the criteria for comparation, a true response is given. | IF <a> <> SET("a","b","c") Means if <a> does not equal a or b or c.  IF <a> = SET(“a”,”b”,”c”) Means if <a> is equal to a or b or c.  IF <a> = SET(0-3) Means if <a> is equal to 0 or 1 or 2 or 3 |
| STARTSWITH | The first characters within a text string field are equal to the stated value. | IF <a> STARTSWITH "T/A" Means the condition is true if field <a> contains a value that starts with the text string ‘T/A’ |
| SUM | The sum of all instance of an element | SUM(<a>) The total value of all instances of <a>, when each <a> is added up. (Where <a> is an element that is part of a repeating tuple or is a repeating element). |
| TEXT() | Used to define the maximum length of a textual field.  (See also LENGTH) | <a> <> TEXT(150) Means the maximum number of characters allowable within field <a> is 150. |
| TUPLE | Concepts that contain a group of two or more fields. Generally, although not always, these concepts are a set of two or more fields that may be repeated, as a group, within a single business document. | TUPLE(addressdetails2.xx.xx:AddressDetails) Means the fields that have been defined as belonging to the ‘addressdetails2.xx.xx:AddressDetails’ module. |
| WHERE | Limits the conditions in a rule to a given context, tuple or condition. | IF period.endDate = NULLORBLANK **WHERE** CONTEXT <> SET(“RPI”, “RPI.Opening”, “RPI.Closing”) Means if the Period End date is not present where the context is anything other than RPI or RPI.Opening, or RPI.Closing.  WHERE IN TUPLE(ptr.0001.lodge.req.xx.xx.StatementOfDistribution) Means the check referred to within the rest of the rule applies where the field(s) exists in the ‘ptr.0001.lodge.req.xx.xx.StatementOfDistribution’ tuple  WHERE(TUPLE ELEMENT pyde.xx.xx:AddressDetails.Usage.Code = "POS") Means the check referred to within the rest of the rule applies where the Address Usage Code in the tuple is equal to ‘POS’. |
| xbrli\ | Indicates the tuple is not within another tuple.  Used where a particular tuple appears more than once within a form or schedule. | IN TUPLE(xbrli\organisationname2.xx.xx:OrganisationNameDetails) Means in the tuple ‘organisationname2.xx.xx:OrganisationNameDetails’ that is not within another tuple.  In this example, the implication is that the ‘organisationname2.xx.xx:OrganisationNameDetails’ also exists under another tuple within the same form or schedule. |

Appendix C – Validation rules alias definitions

| **Validation Rule Alias** | **Fully Defined XBRL Element** |
| --- | --- |
| IEE1 | IEE:RP:pyin.xx.xx:Lodgment.PeriodYear.Year |
| IEE4 | IEE:RP:pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text WHERE ((TUPLE ELEMENT EXPLICIT pyde.xx.xx:OrganisationNameDetails.OrganisationalNameType.Code = "MN") AND (TUPLE ELEMENT EXPLICIT pyde.xx.xx:OrganisationNameDetails.Currency.Code = "C")) IN TUPLE(orgname2.xx.xx:OrganisationNameDetails) |
| IEE5 | IEE:RP:pyde.xx.xx:AddressDetails.Line1.Text WHERE ((TUPLE ELEMENT EXPLICIT pyde.xx.xx:AddressDetails.Usage.Code = "POS") AND (TUPLE ELEMENT EXPLICIT pyde.xx.xx:AddressDetails.Currency.Code = "C")) IN TUPLE(address2.xx.xx:AddressDetails) |
| IEE6 | IEE:RP:pyde.xx.xx:AddressDetails.LocalityName.Text WHERE ((TUPLE ELEMENT EXPLICIT pyde.xx.xx:AddressDetails.Usage.Code = "POS") AND (TUPLE ELEMENT EXPLICIT pyde.xx.xx:AddressDetails.Currency.Code = "C")) IN TUPLE(address2.xx.xx:AddressDetails) |
| IEE10 | IEE:RP:pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text WHERE ((TUPLE ELEMENT EXPLICIT pyde.xx.xx:OrganisationNameDetails.OrganisationalNameType.Code = "MN") AND (TUPLE ELEMENT EXPLICIT pyde.xx.xx:OrganisationNameDetails.Currency.Code = "P")) IN TUPLE(orgname2.xx.xx:OrganisationNameDetails) |
| IEE11 | IEE:RP:pyde.xx.xx:AddressDetails.Line1.Text WHERE ((TUPLE ELEMENT EXPLICIT pyde.xx.xx:AddressDetails.Usage.Code = "POS") AND (TUPLE ELEMENT EXPLICIT pyde.xx.xx:AddressDetails.Currency.Code = "P")) IN TUPLE(address2.xx.xx:AddressDetails) |
| IEE16 | IEE:RP:ptde.xx.xx:Party.Type.Code |
| IEE17 | IEE:RP:pyde.xx.xx:Residency.CentralManagementAndControlOutsideAustralia.Indicator |
| IEE18 | IEE:RP:pyde.xx.xx:Residency.CentralManagementAndControlOutsideAustraliaFullPeriod.Indicator |
| IEE19 | IEE:RP:pyin.xx.xx:PeriodStart.Date WHERE (TUPLE ELEMENT EXPLICIT pyin.xx.xx:Period.Type.Code = "ControlOutsideAustraliaPartPeriod") IN TUPLE(period1.xx.xx:PeriodDetails) |
| IEE20 | IEE:RP:pyin.xx.xx:PeriodEnd.Date WHERE (TUPLE ELEMENT EXPLICIT pyin.xx.xx:Period.Type.Code = "ControlOutsideAustraliaPartPeriod") IN TUPLE(period1.xx.xx:PeriodDetails) |
| IEE21 | IEE:RP:pyde.xx.xx:Residency.TaxPurposesOrganisationStatus.Code |
| IEE22 | IEE:RP:pyde.xx.xx:Residency.NonResidentFullPeriod.Indicator |
| IEE23 | IEE:RP:pyin.xx.xx:PeriodStart.Date WHERE (TUPLE ELEMENT EXPLICIT pyin.xx.xx:Period.Type.Code = "NonResidentPartPeriod") IN TUPLE(period1.xx.xx:PeriodDetails) |
| IEE24 | IEE:RP:pyin.xx.xx:PeriodEnd.Date WHERE (TUPLE ELEMENT EXPLICIT pyin.xx.xx:Period.Type.Code = "NonResidentPartPeriod") IN TUPLE(period1.xx.xx:PeriodDetails) |
| IEE25 | IEE:RP:pyid.xx.xx:Identifiers.TaxFileNumber.Identifier IN TUPLE(iee.0002.lodge.req.xx.xx:TrusteeOrPartner) |
| IEE26 | IEE:RP:pyid.xx.xx:Identifiers.TaxFileNumberInexistent.Indicator IN TUPLE(iee.0002.lodge.req.xx.xx:TrusteeOrPartner) |
| IEE28 | IEE:RP:pyde.xx.xx:PersonNameDetails.FamilyName.Text WHERE (TUPLE ELEMENT EXPLICIT pyde.xx.xx:PersonNameDetails.PersonNameType.Code = "LGL") IN TUPLE(prsnstrcnm3.xx.xx:PersonNameDetails) IN TUPLE(iee.0002.lodge.req.xx.xx:TrusteeOrPartner) |
| IEE32 | IEE:RP:pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text WHERE ((TUPLE ELEMENT EXPLICIT pyde.xx.xx:OrganisationNameDetails.OrganisationalNameType.Code = "MN") AND (TUPLE ELEMENT EXPLICIT pyde.xx.xx:OrganisationNameDetails.Currency.Code = "C")) IN TUPLE(orgname2.xx.xx:OrganisationNameDetails) IN TUPLE(iee.0002.lodge.req.xx.xx:TrusteeOrPartner) |
| IEE34 | IEE:RP:pyid.xx.xx:Identifiers.AustralianCompanyNumber.Identifier IN TUPLE(iee.0002.lodge.req.xx.xx:TrusteeOrPartner) |
| IEE35 | IEE:RP:pyde.xx.xx:AddressDetails.Line1.Text WHERE ((TUPLE ELEMENT EXPLICIT pyde.xx.xx:AddressDetails.Usage.Code = "POS") AND (TUPLE ELEMENT EXPLICIT pyde.xx.xx:AddressDetails.Currency.Code = "C")) IN TUPLE(address2.xx.xx:AddressDetails) IN TUPLE(iee.0002.lodge.req.xx.xx:TrusteeOrPartner) |
| IEE41 | IEE:RP:pyde.xx.xx:Residency.TaxPurposesPersonStatus.Indicator IN TUPLE(iee.0002.lodge.req.xx.xx:TrusteeOrPartner) |
| IEE42 | IEE:RP:pyde.xx.xx:Residency.NonResidentFullPeriod.Indicator IN TUPLE(iee.0002.lodge.req.xx.xx:TrusteeOrPartner) |
| IEE43 | IEE:RP:pyin.xx.xx:PeriodStart.Date WHERE (TUPLE ELEMENT EXPLICIT pyin.xx.xx:Period.Type.Code = "NonResidentPartPeriod") IN TUPLE(period1.xx.xx:PeriodDetails) IN TUPLE(iee.0002.lodge.req.xx.xx:TrusteeOrPartner) |
| IEE44 | IEE:RP:pyin.xx.xx:PeriodEnd.Date WHERE (TUPLE ELEMENT EXPLICIT pyin.xx.xx:Period.Type.Code = "NonResidentPartPeriod") IN TUPLE(period1.xx.xx:PeriodDetails) IN TUPLE(iee.0002.lodge.req.xx.xx:TrusteeOrPartner) |
| IEE45 | IEE:RP:rvctc3.xx.xx:Elections.InterposedEntityElectionStatus.Year |
| IEE46 | IEE:RP:rvctc3.xx.xx:Elections.DateSpecified.Date |
| IEE47 | IEE:RP:pyin.xx.xx:PeriodStart.Date WHERE (TUPLE ELEMENT EXPLICIT pyin.xx.xx:Period.Type.Code = "SubstitutedAccountingPeriod") IN TUPLE(period1.xx.xx:PeriodDetails) |
| IEE48 | IEE:RP:pyin.xx.xx:PeriodEnd.Date WHERE (TUPLE ELEMENT EXPLICIT pyin.xx.xx:Period.Type.Code = "SubstitutedAccountingPeriod") IN TUPLE(period1.xx.xx:PeriodDetails) |
| IEE49 | IEE:RP:rvctc3.xx.xx:Elections.Commencement.Date |
| IEE50 | IEE:RP:pyid.xx.xx:Identifiers.TaxFileNumber.Identifier IN TUPLE(iee.0002.lodge.req.xx.xx:FamilyTrust) |
| IEE51 | IEE:RP:pyid.xx.xx:Identifiers.TaxFileNumberInexistent.Indicator IN TUPLE(iee.0002.lodge.req.xx.xx:FamilyTrust) |
| IEE52 | IEE:RP:OrganisationNameDetails.OrganisationalName.Text WHERE ((TUPLE ELEMENT EXPLICIT pyde.xx.xx:OrganisationNameDetails.OrganisationalNameType.Code = "MN") AND (TUPLE ELEMENT EXPLICIT pyde.xx.xx:OrganisationNameDetails.Currency.Code = "C")) IN TUPLE(orgname2.xx.xx:OrganisationNameDetails) IN TUPLE(iee.0002.lodge.req.xx.xx:FamilyTrust) |
| IEE53 | IEE:RP:pyde.xx.xx:AddressDetails.Line1.Text WHERE ((TUPLE ELEMENT EXPLICIT pyde.xx.xx:AddressDetails.Usage.Code = "POS") AND (TUPLE ELEMENT EXPLICIT pyde.xx.xx:AddressDetails.Currency.Code = "C")) IN TUPLE(address2.xx.xx:AddressDetails) IN TUPLE(iee.0002.lodge.req.xx.xx:FamilyTrust) |
| IEE54 | IEE:RP:pyde.xx.xx:AddressDetails.LocalityName.Text WHERE ((TUPLE ELEMENT EXPLICIT pyde.xx.xx:AddressDetails.Usage.Code = "POS") AND (TUPLE ELEMENT EXPLICIT pyde.xx.xx:AddressDetails.Currency.Code = "C")) IN TUPLE(address2.xx.xx:AddressDetails) IN TUPLE(iee.0002.lodge.req.xx.xx:FamilyTrust) |
| IEE58 | IEE:RP:rvctc3.xx.xx:Elections.FamilyTrustElectionStatus.Year IN TUPLE(iee.0002.lodge.req.xx.xx:FamilyTrust) |
| IEE59 | IEE:RP:rvctc3.xx.xx:Elections.Commencement.Date IN TUPLE(iee.0002.lodge.req.xx.xx:FamilyTrust) |
| IEE60 | IEE:RP:pyin.xx.xx:PeriodStart.Date WHERE (TUPLE ELEMENT EXPLICIT pyin.xx.xx:Period.Type.Code = "SubstitutedAccountingPeriod") IN TUPLE(period1.xx.xx:PeriodDetails) IN TUPLE(iee.0002.lodge.req.xx.xx:FamilyTrust) |
| IEE61 | IEE:RP:pyin.xx.xx:PeriodEnd.Date WHERE (TUPLE ELEMENT EXPLICIT pyin.xx.xx:Period.Type.Code = "SubstitutedAccountingPeriod") IN TUPLE(period1.xx.xx:PeriodDetails) IN TUPLE(iee.0002.lodge.req.xx.xx:FamilyTrust) |
| IEE62 | IEE:RP:pyid.xx.xx:Identifiers.TaxFileNumber.Identifier IN TUPLE(iee.0002.lodge.req.xx.xx:SpecifiedIndividual) |
| IEE63 | IEE:RP:pyid.xx.xx:Identifiers.TaxFileNumberInexistent.Indicator IN TUPLE(iee.0002.lodge.req.xx.xx:SpecifiedIndividual) |
| IEE64 | IEE:RP:pyde.xx.xx:PersonDemographicDetails.Birth.Date IN TUPLE(iee.0002.lodge.req.xx.xx:SpecifiedIndividual) |
| IEE65 | IEE:RP:pyde.xx.xx:PersonNameDetails.Title.Text WHERE (TUPLE ELEMENT EXPLICIT pyde.xx.xx:PersonNameDetails.PersonNameType.Code = "LGL") IN TUPLE(prsnstrcnm3.xx.xx:PersonNameDetails) IN TUPLE(iee.0002.lodge.req.xx.xx:SpecifiedIndividual) |
| IEE66 | IEE:RP:pyde.xx.xx:PersonNameDetails.FamilyName.Text WHERE (TUPLE ELEMENT EXPLICIT pyde.xx.xx:PersonNameDetails.PersonNameType.Code = "LGL") IN TUPLE(prsnstrcnm3.xx.xx:PersonNameDetails) IN TUPLE(iee.0002.lodge.req.xx.xx:SpecifiedIndividual) |
| IEE67 | IEE:RP:pyde.xx.xx:PersonNameDetails.NameSuffix.Text WHERE (TUPLE ELEMENT EXPLICIT pyde.xx.xx:PersonNameDetails.PersonNameType.Code = "LGL") IN TUPLE(prsnstrcnm3.xx.xx:PersonNameDetails) IN TUPLE(iee.0002.lodge.req.xx.xx:SpecifiedIndividual) |
| IEE68 | IEE:RP:pyde.xx.xx:PersonNameDetails.GivenName.Text WHERE (TUPLE ELEMENT EXPLICIT pyde.xx.xx:PersonNameDetails.PersonNameType.Code = "LGL") IN TUPLE(prsnstrcnm3.xx.xx:PersonNameDetails) IN TUPLE(iee.0002.lodge.req.xx.xx:SpecifiedIndividual) |
| IEE69 | IEE:RP:pyde.xx.xx:PersonNameDetails.OtherGivenName.Text WHERE (TUPLE ELEMENT EXPLICIT pyde.xx.xx:PersonNameDetails.PersonNameType.Code = "LGL") IN TUPLE(prsnstrcnm3.xx.xx:PersonNameDetails) IN TUPLE(iee.0002.lodge.req.xx.xx:SpecifiedIndividual) |
| IEE70 | IEE:RP:pyde.xx.xx:AddressDetails.Line1.Text WHERE ((TUPLE ELEMENT EXPLICIT pyde.xx.xx:AddressDetails.Usage.Code = "RES") AND (TUPLE ELEMENT EXPLICIT pyde.xx.xx:AddressDetails.Currency.Code = "C")) IN TUPLE(address2.xx.xx:AddressDetails) IN TUPLE(iee.0002.lodge.req.xx.xx:SpecifiedIndividual) |
| IEE71 | IEE:RP:pyde.xx.xx:AddressDetails.LocalityName.Text WHERE ((TUPLE ELEMENT EXPLICIT pyde.xx.xx:AddressDetails.Usage.Code = "RES") AND (TUPLE ELEMENT EXPLICIT pyde.xx.xx:AddressDetails.Currency.Code = "C")) IN TUPLE(address2.xx.xx:AddressDetails) IN TUPLE(iee.0002.lodge.req.xx.xx:SpecifiedIndividual) |
| IEE72 | IEE:RP:pyde.xx.xx:AddressDetails.StateOrTerritory.Code WHERE ((TUPLE ELEMENT EXPLICIT pyde.xx.xx:AddressDetails.Usage.Code = "RES") AND (TUPLE ELEMENT EXPLICIT pyde.xx.xx:AddressDetails.Currency.Code = "C")) IN TUPLE(address2.xx.xx:AddressDetails) IN TUPLE(iee.0002.lodge.req.xx.xx:SpecifiedIndividual) |
| IEE73 | IEE:RP:pyde.xx.xx:AddressDetails.Postcode.Text WHERE ((TUPLE ELEMENT EXPLICIT pyde.xx.xx:AddressDetails.Usage.Code = "RES") AND (TUPLE ELEMENT EXPLICIT pyde.xx.xx:AddressDetails.Currency.Code = "C")) IN TUPLE(address2.xx.xx:AddressDetails) IN TUPLE(iee.0002.lodge.req.xx.xx:SpecifiedIndividual) |
| IEE82 | IEE:RP:rvctc3.xx.xx:Elections.Revocation.Year |
| IEE88 | IEE:RP:OrganisationNameDetails.OrganisationalName.Text WHERE ((TUPLE ELEMENT EXPLICIT pyde.xx.xx:OrganisationNameDetails.OrganisationalNameType.Code = "MN") AND (TUPLE ELEMENT EXPLICIT pyde.xx.xx:OrganisationNameDetails.Currency.Code = "C")) IN TUPLE(orgname2.xx.xx:OrganisationNameDetails) IN TUPLE(iee.0002.lodge.req.xx.xx:DeclarationOfTrusteeOrPartner) |
| IEE95 | IEE:RP:pyid.xx.xx:Identifiers.AustralianRegisteredBodyNumber.Identifier IN TUPLE(iee.0002.lodge.req.xx.xx:TrusteeOrPartner) |
| IEE100 | IEE:RP:pyid.xx.xx:Identifiers.AustralianCompanyNumber.Identifier |
| IEE101 | IEE:RP:pyid.xx.xx:Identifiers.AustralianRegisteredBodyNumber.Identifier |
| IEE116 | IEE:RP:rvctc3.xx.xx:Elections.InterposedEntityElectionOrRevocation.Code |
| IEE119 | IEE:RP:OrganisationNameDetails.OrganisationNameType.Code WHERE (TUPLE ELEMENT EXPLICIT pyde.xx.xx:OrganisationNameDetails.Currency.Code = “C”) IN TUPLE(orgname2.xx.xx:OrganisationNameDetails) IN TUPLE(iee.0002.lodge.req.xx.xx:FamilyTrust) |
| IEE146 | IEE:RP:pyin.xx.xx:Declaration.StatementAccepted.Indicator IN TUPLE(declaration2.xx.xx:Declaration) IN TUPLE(iee.0002.lodge.req.xx.xx:DeclarationOfTrusteeOrPartner) |
| IEE147 | IEE:RP:pyin.xx.xx:Declaration.StatementType.Code IN TUPLE(declaration2.xx.xx:Declaration) |
| IEE159 | IEE:RP:pyin.xx.xx:Declaration.StatementType.Code IN TUPLE(declaration2.xx.xx:Declaration) |
| IEE170 | IEE:RP:OrganisationNameDetails.Currency.Code WHERE (TUPLE ELEMENT EXPLICIT pyde.xx.xx:OrganisationNameDetails.OrganisationalNameType.Code = “MN”) IN TUPLE(orgname2.xx.xx:OrganisationNameDetails) IN TUPLE(iee.0002.lodge.req.xx.xx:FamilyTrust) |
| IEE179 | IEE:INT:pyin.xx.xx:Declaration.StatementAccepted.Indicator IN TUPLE(declaration2.xx.xx:Declaration) |
| IEE185 | IEE:INT:pyde.xx.xx:PersonUnstructuredName.Usage.Code IN TUPLE(prsnunstrcnm1.xx.xx:PersonUnstructuredName) |
| IEE186 | IEE:INT:pyde.xx.xx:PersonUnstructuredName.FullName.Text WHERE (TUPLE ELEMENT EXPLICIT pyde.xx.xx:PersonUnstructuredName.Usage.Code = “Contact”) IN TUPLE(prsnunstrcnm1.xx.xx:PersonUnstructuredName) |
| IEE192 | IEE:RP:pyin.xx.xx:Declaration.StatementAccepted.Indicator IN TUPLE(declaration2.xx.xx:Declaration) |
| IEE193 | IEE:RP:pyin.xx.xx:Declaration.StatementType.Code IN TUPLE(declaration2.xx.xx:Declaration) IN TUPLE(iee.0002.lodge.req.xx.xx:DeclarationOfTrusteeOrPartner) |
| IEE203 | IEE:RP:pyde.xx.xx:PersonNameDetails.FamilyName.Text WHERE ((TUPLE ELEMENT EXPLICIT pyde.xx.xx:PersonNameDetails.PersonNameType.Code = "LGL") AND (TUPLE ELEMENT EXPLICIT pyde.xx.xx:PersonNameDetails.Usage.Code = "DeclarationSignatory") AND (TUPLE ELEMENT EXPLICIT pyde.xx.xx:PersonNameDetails.Currency.Code = "C")) IN TUPLE(prsnstrcnm3.xx.xx:PersonNameDetails) IN TUPLE(iee.0002.lodge.req.xx.xx:DeclarationOfTrusteeOrPartner) |

Appendix D – Differences between the IEE standalone form and the IEE schedule

The IEE.LODGE request message content table (5.5.1.3 above) lists all the facts that apply to an IEE. Some facts, however, will apply only when an IEE is lodged as a stand-alone form, others will apply only when lodged as a schedule accompanying a main form. This section summarises those differences.

* A revocation (as opposed to an election) may be made only using the schedule.
* The name and address of the trust – both current and previous - is submitted as part of the main form when the IEE is lodged as a schedule. When lodged as a stand-alone form, the name and address must be included as a part of the IEE.

As a consequence of these differences, some validations rules apply to only the schedule or only to the form.

**Validation rules that apply to the schedule only**

* VR.ATO.IEE.402011
* VR.ATO.IEE.410051
* VR.ATO.IEE.410069
* VR.ATO.IEE.410070
* VR.ATO.IEE.410071
* VR.ATO.IEE.410076
* VR.ATO.IEE.410084
* VR.ATO.IEE.410100
* VR.ATO.GEN.402010
* VR.ATO.GEN.410115
* VR.ATO.GEN.410119
* VR.ATO.IEE.410193
* VR.ATO.IEE.410198
* VR.ATO.IEE.410223
* VR.ATO.IEE.410224
* VR.ATO.IEE.410225
* VR.ATO.IEE.410226
* VR.ATO.IEE.410262

**Validation rules that apply to the stand-alone form only**

* VR.ATO.IEE.410041
* VR.ATO.IEE.410086
* VR.ATO.IEE.410099
* VR.ATO.IEE.410135
* VR.ATO.IEE.410136
* VR.ATO.IEE.410137
* VR.ATO.IEE.410197
* VR.ATO.IEE.410199
* VR.ATO.IEE.410261

Appendix E – Common Validation Rules

**E.1 - Common Address rules**

| **XBRL Fact** | **Common Address Rules** | **Rule Implementation** | **SBR Message Code** |
| --- | --- | --- | --- |
| AddressDetails.OverseasAddress.Indicator | WHERE IN TUPLE(address2.xx.xx:AddressDetails) IF (pyde.xx.xx:AddressDetails.OverseasAddress.Indicator = NULLORBLANK)  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410167 | CMN.ATO.GEN.410167 |
| pyde.xx.xx:AddressDetails.Line1.Text | WHERE IN TUPLE(address2.xx.xx:AddressDetails) IF (pyde.xx.xx:AddressDetails.Line1.Text = NULLORBLANK)  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410214 | CMN.ATO.GEN.410214 |
| pyde.xx.xx:AddressDetails.Country.Code | WHERE IN TUPLE(address2.xx.xx:AddressDetails) IF (pyde.xx.xx:AddressDetails.Country.Code <> NULLORBLANK) AND (pyde.xx.xx:AddressDetails.Country.Code <> "au") AND (pyde.xx.xx:AddressDetails.OverseasAddress.Indicator = FALSE)   RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410212 | CMN.ATO.GEN.410212 |
| pyde.xx.xx:AddressDetails.Country.Code | WHERE IN TUPLE(address2.xx.xx:AddressDetails) IF (pyde.xx.xx:AddressDetails.OverseasAddress.Indicator = TRUE) AND (pyde.xx.xx:AddressDetails.Country.Code = NULLORBLANK)  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410191 | CMN.ATO.GEN.410191 |
| pyde.xx.xx:AddressDetails.Country.Code | WHERE IN TUPLE(address2.xx.xx:AddressDetails) IF (pyde.xx.xx:AddressDetails.Country.Code <> NULLORBLANK) AND (pyde.xx.xx:AddressDetails.Country.Code <> SET(DOMAIN(COUNTRY CODES)))  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410192 | CMN.ATO.GEN.410192 |
| pyde.xx.xx:AddressDetails.CountryName.Text | IF (pyde.xx.xx:AddressDetails.CountryName.Text <> NULLORBLANK)  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410213 | CMN.ATO.GEN.410213 |
| pyde.xx.xx:AddressDetails.Currency.Code | IN TUPLE(address2.xx.xx:AddressDetails) IF pyde.xx.xx:AddressDetails.Currency.Code = NULLORBLANK  RETURN VALIDATION MESSAGE END IF | VR.ATO.GEN.410148 | CMN.ATO.GEN.410148 |
| pyde.xx.xx:AddressDetails.Currency.Code | IF (pyde.xx.xx:AddressDetails.Currency.Code <> NULLORBLANK) AND (pyde.xx.xx:AddressDetails.Currency.Code <> SET("C","P"))  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.428202 | CMN.ATO.GEN.428202 |
| pyde.xx.xx:AddressDetails.Line1.Text | IF (pyde.xx.xx:AddressDetails.Line1.Text CONTAINS "UNKNOWN")  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.000405 | CMN.ATO.GEN.000405 |
| pyde.xx.xx:AddressDetails.Line1.Text | IF (pyde.xx.xx:AddressDetails.Line1.Text = FOUND("C/O","C/","Care Of","CO"))  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN. 428240 | CMN.ATO.GEN.000406 |
| pyde.xx.xx:AddressDetails.Line1.Text | WHERE IN TUPLE(address2.xx.xx:AddressDetails) IF ((pyde.xx.xx:AddressDetails.Line1.Text CONTAINS "C/-") AND (pyde.xx.xx:AddressDetails.Line2.Text = NULLORBLANK))  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410002 | CMN.ATO.GEN.410002 |
| pyde.xx.xx:AddressDetails.Line1.Text | WHERE IN TUPLE(address2.xx.xx:AddressDetails) IF ((pyde.xx.xx:AddressDetails.StateOrTerritory.Code <> NULLORBLANK) OR (pyde.xx.xx:AddressDetails.Postcode.Text <> NULLORBLANK)) AND (pyde.xx.xx:AddressDetails.Locality.Text = NULLORBLANK)  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410003 | CMN.ATO.GEN.410003 |
| pyde.xx.xx:AddressDetails.Line1.Text | WHERE IN TUPLE(address2.xx.xx:AddressDetails) IF (pyde.xx.xx:AddressDetails.Line1.Text <> NULLORBLANK) AND ((pyde.xx.xx:AddressDetails.StateOrTerritory.Code = NULLORBLANK) AND (pyde.xx.xx:AddressDetails.Postcode.Text = NULLORBLANK) AND (pyde.xx.xx:AddressDetails.Country.Code = NULLORBLANK))  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410053 | CMN.ATO.GEN.410053 |
| pyde.xx.xx:AddressDetails.Line1.Text | IF (pyde.xx.xx:AddressDetails.Line1.Text = FOUND("AS ABOVE"))  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410205 | CMN.ATO.GEN.410001 |
| pyde.xx.xx:AddressDetails.Line1.Text | IF (pyde.xx.xx:AddressDetails.Line1.Text > LENGTH(38))  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410194 | CMN.ATO.GEN.410194 |
| pyde.xx.xx:AddressDetails.Line2.Text | IF (pyde.xx.xx:AddressDetails.Line2.Text CONTAINS "UNKNOWN")  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.000407 | CMN.ATO.GEN.000407 |
| pyde.xx.xx:AddressDetails.Line2.Text | IF (pyde.xx.xx:AddressDetails.Line2.Text = FOUND("C/-","C/O","C/","Care Of","CO"))  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.428241 | CMN.ATO.GEN.000408 |
| pyde.xx.xx:AddressDetails.Line2.Text | IF (pyde.xx.xx:AddressDetails.Line2.Text > LENGTH(38))  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410195 | CMN.ATO.GEN.410195 |
| pyde.xx.xx:AddressDetails.Line3.Text | IF (pyde.xx.xx:AddressDetails.Line3.Text <> NULLORBLANK)  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410008 | CMN.ATO.GEN.410008 |
| pyde.xx.xx:AddressDetails.Line4.Text | IF (pyde.xx.xx:AddressDetails.Line4.Text <> NULLORBLANK)  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410013 | CMN.ATO.GEN.410013 |
| pyde.xx.xx:AddressDetails.LocalityName.Text | IF (pyde.xx.xx:AddressDetails.LocalityName.Text CONTAINS "UNKNOWN")  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.000409 | CMN.ATO.GEN.000409 |
| pyde.xx.xx:AddressDetails.LocalityName.Text | IF (pyde.xx.xx:AddressDetails.LocalityName.Text CONTAINS NUMERICS SET(0-9))  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.000410 | CMN.ATO.GEN.000410 |
| pyde.xx.xx:AddressDetails.LocalityName.Text | IF (pyde.xx.xx:AddressDetails.LocalityName.Text = FOUND("QLD","NSW”,"VIC","SA","WA","NT","ACT","TAS"))  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.428230 | CMN.ATO.GEN.000411 |
| pyde.xx.xx:AddressDetails.OverseasAddress.Indicator | WHERE IN TUPLE(address2.xx.xx:AddressDetails) IF (pyde.xx.xx:AddressDetails.Country.Code = "au") AND (pyde.xx.xx:AddressDetails.OverseasAddress.Indicator = TRUE)   RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410211 | CMN.ATO.GEN.410211 |
| pyde.xx.xx:AddressDetails.Postcode.Text | IF (pyde.xx.xx:AddressDetails.Postcode.Text <> NULLORBLANK) AND (pyde.xx.xx:AddressDetails.Postcode.Text <> SET(0200-9999))  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.000460 | CMN.ATO.GEN.000460 |
| pyde.xx.xx:AddressDetails.Postcode.Text | WHERE IN TUPLE(address2.xx.xx:AddressDetails) IF (pyde.xx.xx:AddressDetails.StateOrTerritory.Code <> NULLORBLANK) AND (pyde.xx.xx:AddressDetails.Postcode.Text = NULLORBLANK)  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410055 | CMN.ATO.GEN.410055 |
| pyde.xx.xx:AddressDetails.StateOrTerritory.Code | IF (pyde.xx.xx:AddressDetails.StateOrTerritory.Code <> NULLORBLANK) AND (pyde.xx.xx:AddressDetails.StateOrTerritory.Code <> SET("NSW","VIC","ACT","QLD","TAS","SA","WA","NT"))  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.001007 | CMN.ATO.GEN.001007 |
| pyde.xx.xx:AddressDetails.StateOrTerritory.Code | WHERE IN TUPLE(address2.xx.xx:AddressDetails) IF (pyde.xx.xx:AddressDetails.Postcode.Text <> NULLORBLANK) AND (pyde.xx.xx:AddressDetails.StateOrTerritory.Code = NULLORBLANK)  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410017 | CMN.ATO.GEN.410017 |
| pyde.xx.xx:AddressDetails.StateOrTerritory.Code | WHERE IN TUPLE(address2.xx.xx:AddressDetails) IF (pyde.xx.xx:AddressDetails.StateOrTerritory.Code = NULLORBLANK) AND (pyde.xx.xx:AddressDetails.Postcode.Text = NULLORBLANK) AND (pyde.xx.xx:AddressDetails.Country.Code = NULLORBLANK)  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410018 | CMN.ATO.GEN.410018 |
| pyde.xx.xx:AddressDetails.Usage.Code | WHERE IN TUPLE(address2.xx.xx:AddressDetails) IF (pyde.xx.xx:AddressDetails.Usage.Code = NULLORBLANK)   RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410174 | CMN.ATO.GEN.410174 |

**E.2 - Common Organisation Name rules**

| **XBRL Fact** | **Common Organisation Name Rules** | **Rule Implementation** | **SBR Message Code** | |
| --- | --- | --- | --- | --- |
| pyde.xx.xx:OrganisationNameDetails.Currency.Code | IF pyde.xx.xx:OrganisationNameDetails.Currency.Code <> NULLORBLANK AND pyde.xx.xx:OrganisationNameDetails.Currency.Code <> SET ("C", "P")  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.000466 | CMN.ATO.GEN.000466 | |
| pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text | IF (pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text CONTAINS " - ")  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.000413 | CMN.ATO.GEN.000413 | |
| pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text | IF (pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text CONTAINS "P/L")  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.000414 | CMN.ATO.GEN.000414 | |
| pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text | IF (pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text = FOUND("The trustee","The Exec","exec","The TTE"))  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410206 | CMN.ATO.GEN.428042 | |
| pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text | IF (pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text CONTAINS SET("--", """", " "))  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.000416 | CMN.ATO.GEN.000416 | |
| pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text | IF (pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text ENDSWITH SET(" T/A"," T/A P"ship"," T/A Pship"," T/A P/Ship"," T/A Partnership"))  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.000417 | CMN.ATO.GEN.000417 | |
| pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text | IF (pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text STARTSWITH "T/A ") AND (pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text ENDSWITH SET(" Pship"," P'ship"," P/ship"))  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410038 | CMN.ATO.GEN.410038 | |
| pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text | IF (pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text <> NULLORBLANK) AND (pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text DOES NOT CONTAIN SET("a-z", "A-Z", "0-9"))  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410039 | CMN.ATO.GEN.410039 | |
| pyde.xx.xx:OrganisationNameDetails.OrganisationalNameType.Code | WHERE IN TUPLE(orgname2.xx.xx:OrganisationNameDetails) IF (pyde.xx.xx:OrganisationNameDetails.OrganisationalNameType.Code = NULLORBLANK)   RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410147 | CMN.ATO.GEN.410147 | |
| pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text | WHERE IN TUPLE(orgname2.xx.xx:OrganisationNameDetails) IF (pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text = NULLORBLANK)  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410215 | | CMN.ATO.GEN.410215 | |
| pyde.xx.xx:OrganisationNameDetails.Currency.Code | IN TUPLE(orgname2.xx.xx:OrganisationNameDetails) IF (pyde.xx.xx:OrganisationNameDetails.Currency.Code = NULLORBLANK)  RETURN VALIDATION MESSAGE ENDIF | VR.ATO.GEN.410140 | | CMN.ATO.GEN.410140 | |

**E.3 - Common Person Name rules**

| **XBRL Fact** | **Common Person Name Rules** | **Rule Implementation** | **SBR Message Code** |
| --- | --- | --- | --- |
| pyde.xx.xx:PersonNameDetails.PersonNameType.Code | WHERE IN TUPLE(prsnstrcnm3.xx.xx:PersonNameDetails) IF (pyde.xx.xx:PersonNameDetails.PersonNameType.Code = NULLORBLANK)   RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410216 | CMN.ATO.GEN.410216 |
| pyde.xx.xx:PersonNameDetails.FamilyName.Text | WHERE IN TUPLE(prsnstrcnm3.xx.xx:PersonNameDetails) IF (pyde.xx.xx:PersonNameDetails.FamilyName.Text = NULLORBLANK)   RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410217 | CMN.ATO.GEN.410217 |
| pyde.xx.xx:PersonNameDetails.FamilyName.Text | IF (pyde.xx.xx:PersonNameDetails.FamilyName.Text = FOUND("ESQ","II","III","IV","JNR","JP","MHA","MHR","MLA","MLC","MP","QC","SNR"))  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.428231 | CMN.ATO.GEN.000422 |
| pyde.xx.xx:PersonNameDetails.FamilyName.Text | IF (pyde.xx.xx:PersonNameDetails.FamilyName.Text CONTAINS " - ")  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.000423 | CMN.ATO.GEN.000423 |
| pyde.xx.xx:PersonNameDetails.FamilyName.Text | IF (pyde.xx.xx:PersonNameDetails.FamilyName.Text = FOUND("Exec for","Rep for","Trustee for"))  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410200 | CMN.ATO.GEN.000424 |
| pyde.xx.xx:PersonNameDetails.FamilyName.Text | IF (pyde.xx.xx:PersonNameDetails.FamilyName.Text = FOUND("MR","MRS","MISS","MS"))  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410201 | CMN.ATO.GEN.000426 |
| pyde.xx.xx:PersonNameDetails.FamilyName.Text | IF (pyde.xx.xx:PersonNameDetails.FamilyName.Text CONTAINS SET("--", """", " "))  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.000427 | CMN.ATO.GEN.000427 |
| pyde.xx.xx:PersonNameDetails.FamilyName.Text | IF (pyde.xx.xx:PersonNameDetails.FamilyName.Text <> NULLORBLANK) AND (pyde.xx.xx:PersonNameDetails.FamilyName.Text DOES NOT CONTAIN SET("A-Z","a-z"))  RETURN VALIDATION MESSSAGE ENDIF | Schematron ID = VR.ATO.GEN.410040 | CMN.ATO.GEN.410040 |
| pyde.xx.xx:PersonNameDetails.GivenName.Text | IF (pyde.xx.xx:PersonNameDetails.GivenName.Text CONTAINS " - ")  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.000434 | CMN.ATO.GEN.000434 |
| pyde.xx.xx:PersonNameDetails.GivenName.Text | IF (pyde.xx.xx:PersonNameDetails.GivenName.Text = FOUND("ESQ","II","III","IV","JNR","JP","MHA","MHR","MLA","MLC","MP","QC","SNR"))  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410202 | CMN.ATO.GEN.000436 |
| pyde.xx.xx:PersonNameDetails.GivenName.Text | IF (pyde.xx.xx:PersonNameDetails.GivenName.Text = FOUND("Exec for","Rep for","Trustee for"))  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410203 | CMN.ATO.GEN.000437 |
| pyde.xx.xx:PersonNameDetails.GivenName.Text | IF (pyde.xx.xx:PersonNameDetails.GivenName.Text = FOUND("MR","MRS","MISS","MS"))  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410204 | CMN.ATO.GEN.000438 |
| pyde.xx.xx:PersonNameDetails.GivenName.Text | IF (pyde.xx.xx:PersonNameDetails.GivenName.Text CONTAINS SET("--", """", " "))  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.000439 | CMN.ATO.GEN.000439 |
| pyde.xx.xx:PersonNameDetails.GivenName.Text | IF (pyde.xx.xx:PersonNameDetails.GivenName.Text <> NULLORBLANK) AND (pyde.xx.xx:PersonNameDetails.GivenName.Text DOES NOT CONTAIN SET("A-Z","a-z"))  RETURN VALIDATION MESSSAGE ENDIF | Schematron ID = VR.ATO.GEN.410063 | CMN.ATO.GEN.410063 |
| pyde.xx.xx:PersonNameDetails.NameSuffix.Text | IF (pyde.xx.xx:PersonNameDetails.NameSuffix.Text <> NULLORBLANK) AND (pyde.xx.xx:PersonNameDetails.NameSuffix.Text <> SET(DOMAIN(SUFFIX CODES)))  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.000458 | CMN.ATO.GEN.000458 |
| pyde.xx.xx:PersonNameDetails.OtherGivenName.Text | IN TUPLE(prsnstrcnm3.xx.xx:PersonNameDetails)  IF (pyde.xx.xx:PersonNameDetails.OtherGivenName.Text <> NULLORBLANK) AND (pyde.xx.xx:PersonNameDetails.GivenName.Text = NULLORBLANK)   RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410218 | CMN.ATO.GEN.410218 |
| pyde.xx.xx:PersonNameDetails.OtherGivenName.Text | IF (pyde.xx.xx:PersonNameDetails.OtherGivenName.Text CONTAINS " - ")  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.000446 | CMN.ATO.GEN.000446 |
| pyde.xx.xx:PersonNameDetails.OtherGivenName.Text | IF (pyde.xx.xx:PersonNameDetails.OtherGivenName.Text = FOUND("ESQ","II","III","IV","JNR","JP","MHA","MHR","MLA","MLC","MP","QC","SNR"))  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410207 | CMN.ATO.GEN.000448 |
| pyde.xx.xx:PersonNameDetails.OtherGivenName.Text | IF (pyde.xx.xx:PersonNameDetails.OtherGivenName.Text = FOUND("Exec for","Rep for","Trustee for"))  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410208 | CMN.ATO.GEN.000449 |
| pyde.xx.xx:PersonNameDetails.OtherGivenName.Text | IF (pyde.xx.xx:PersonNameDetails.OtherGivenName.Text = FOUND("MR","MRS","MISS","MS"))  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.410209 | CMN.ATO.GEN.000450 |
| pyde.xx.xx:PersonNameDetails.OtherGivenName.Text | IF (pyde.xx.xx:PersonNameDetails.OtherGivenName.Text CONTAINS SET("--", """", " "))  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.000451 | CMN.ATO.GEN.000451 |
| pyde.xx.xx:PersonNameDetails.OtherGivenName.Text | IF (pyde.xx.xx:PersonNameDetails.OtherGivenName.Text <> NULLORBLANK) AND (pyde.xx.xx:PersonNameDetails.OtherGivenName.Text DOES NOT CONTAIN SET("A-Z","a-z"))  RETURN VALIDATION MESSSAGE ENDIF | Schematron ID = VR.ATO.GEN.410131 | CMN.ATO.GEN.410131 |
| pyde.xx.xx:PersonNameDetails.Title.Text | IF (pyde.xx.xx:PersonNameDetails.Title.Text <> NULLORBLANK) AND (pyde.xx.xx:PersonNameDetails.Title.Text <> SET(DOMAIN(TITLE CODES)))  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.000459 | CMN.ATO.GEN.000459 |
| pyde.xx.xx:PersonNameDetails.Currency.Code | IN TUPLE(prsnstrcnm3.xx.xx:PersonNameDetails) IF (pyde.xx.xx:PersonNameDetails.Currency.Code = NULLORBLANK)  RETURN VALIDATION MESSAGE END IF | VR.ATO.GEN.410183 | CMN.ATO.GEN.410183 |
| pyde.xx.xx:PersonNameDetails.Usage.Code | IN TUPLE(prsnstrcnm3.xx.xx:PersonNameDetails) IF (pyde.xx.xx:PersonNameDetails.Usage.Code = NULLORBLANK)  RETURN VALIDATION MESSAGE END IF | VR.ATO.GEN.410180 | CMN.ATO.GEN.410180 |

Appendix F – Domain Definitions

**DOMAIN(COUNTRY CODES)**

The domain of valid Country codes is defined in the SBR Definitional Taxonomy for the element pyde.xx.xx:AddressDetails.Country.Code

**DOMAIN(TITLE CODES)**

| **Code** | **Title** | **Code** | **Title** |
| --- | --- | --- | --- |
| 2LT | Second Lieutenant | LBDR | Lance Bombardier |
| AB | Able Seaman | LCPL | Lance Corporal |
| ABBOT | Abbot | LORD | Lord |
| AC | Airman/Aircraftman | LS | Leading Seaman |
| ACM | Air Chief Marshal | LT | Lieutenant |
| ADM | Admiral | LT CMDR | Lieutenant Commander |
| AIR CDRE | Air Commodore | LT COL | Lieutenant Colonel |
| ALDERMAN | Alderman | LT GEN | Lieutenant General |
| AM | Air Marshal | MADAM | Madam |
| ARCHBISHOP | Archbishop | MAJ | Major |
| ARCHDEACON | Archdeacon | MAJ GEN | Major General |
| ASSOC PROF | Associate Professor | MASTER | Master |
| AVM | Air Vice Marshal | MATRON | Matron |
| BARON | Baron | MAYOR | Mayor |
| BARONESS | Baroness | MAYORESS | Mayoress |
| BISHOP | Bishop | MIDN | Midshipman |
| BR | Brother | MISS | Miss |
| BRIG | Brigadier | MON | Monsignor |
| CANON | Canon | MOST REV | Most Reverend |
| CAPT | Captain | MR | Mr |
| CARDINAL | Cardinal | MRS | Mrs |
| CDRE | Commodore | MS | Ms |
| CDT | Cadet | PASTOR | Pastor |
| CHAP | Chaplain | PATRIARCH | Patriarch |
| CMDR | Commander | PLT OFF | Pilot Officer |
| CMM | Commissioner | PO | Petty Officer |
| COL | Colonel | PRIOR | Prior |
| CONST | Constable | PROF | Professor |
| COUNT | Count | PTE | Private |
| COUNTESS | Countess | RABBI | Rabbi |
| CPL | Corporal | RADM | Rear Admiral |
| CPO | Chief Petty Officer | RECTOR | Rector |
| DAME | Dame | REV | Reverend |
| DEACON | Deacon | RF | Representative for |
| DEACONESS | Deaconess | RT HON | Right Honourable |
| DEAN | Dean | RT REV | Right Reverend |
| DEPUTY SUPT | Deputy Superintendent | RT REV BISHOP | Right Reverend Bishop |
| DR | Doctor | RT REV MON | Right Reverend Monsignor |
| DUCHESS | Duchess | SBLT | Sub Lieutenant |
| DUKE | Duke | SEN | Senator |
| EARL | Earl | SGT | Sergeant |
| EF | Executor for | SIR | Sir |
| FLGOFF | Flying Officer | SMN | Seaman |
| FLT LT | Flight Lieutenant | SNR CONST | Senior Constable |
| FR | Father | SQN LDR | Squadron Leader |
| FSGT | Flight Sergeant | SR | Sister |
| GEN | General | SSGT | Staff Sergeant |
| GNR | Gunner | SUPR | Superintendent |
| GP CAPT | Group Captain | SWAMI | Swami |
| HON | Honourable | TF | Trustee for |
| HON JUDGE | Honourable Judge | VADM | Vice Admiral |
| HON JUST | Honourable Justice | VERY REV | Very Reverend |
| HRH | His/Her Royal Highness | VICAR | Vicar |
| INSP | Inspector | VISCOUNT | Viscount |
| JUDGE | Judge | WG CDR | Wing Commander |
| JUST | Justice | WO | Warrant Officer |
| LAC | Leading Aircraftman | WO1 | Warrant Officer Class 1 |
| LACW | Leading Aircraftwoman | WO2 | Warrant Officer Class 2 |
| LADY | Lady |  |  |

**DOMAIN(SUFFIX CODES)**

|  |  |
| --- | --- |
| **Code** | **Suffix** |
| ESQ | Esquire |
| II | The Second |
| III | The Third |
| IV | The Fourth |
| JNR | Junior |
| JP | Justice of the Peace |
| MHA | Member House of Assembly |
| MHR | Member House of Representatives |
| MLA | Member Legislative Assembly |
| MLC | Member Legislative Council |
| MP | Member of Parliament |
| QC | Queens Council |
| SNR | Senior |