|  |  |  |  |
| --- | --- | --- | --- |
|  | | | |
|  | |  |  |
| Standard Business Reporting  *Australian Taxation Office - Dividends and Interest Schedule (dis.0001)*  Message Implementation Guide  Program name: *Standard Business Reporting*  Date: 21th September 2012  Production Release – 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 | 4/03/2010 | Initial release |
| 1.1 | 29/03/2010 | PCR100: Validation rule changes and changes to the context naming to reflect taxonomy naming changes |
| 1.2 | 22/04/2010 | PCR100: Updated “Appendix D” to reflect changes / confirmation of standard name and address rules  Updated TOC  Updated “Domain Definitions” from “Appendix D” to “Appendix E” – duplicate title |
| 1.3a | 22/07/2010 | Replaced rule VR.ATO.GEN.000020 with VR.ATO.GEN.428040  Updated XBRL element names for DIS18 and DIS19 in Appendix C  Updated Table 5.2.1.3.2.1 to show that the tuple dis.0001.lodge.req.xx.xx:UnfrankedDividends is contained within tuple dis.0001.lodge.req.xx.xx:Investor, and contains two elements that are reported in different contexts |
| 1.4 | 26/07/2012 | Added full list of Alias definitions  Added new rules: VR.ATO.DIS.406023, VR.ATO.DIS.406024, VR.ATO.DIS.406025  Modified rules: VR.ATO.DIS.406010, VR.ATO.DIS.406015, VR.ATO.DIS.406016, VR.ATO.DIS.406021, VR.ATO.DIS.406022  Please refer to MIG v1.3 Errata documentation for specific change detail. |
| 1.5 | 21/09/2012 | Updated DIS LODGE Request Message Content table to more clearly specify how the UnfrankedDividend Tuple should be implemented in the XBRL message. |

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

Copyright

© Commonwealth of Australia 2012

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

[1 Introduction 6](#_Toc328413436)

[1.1 Purpose 6](#_Toc328413437)

[1.2 Audience and Scope 6](#_Toc328413438)

[1.3 References 6](#_Toc328413439)

[1.4 Change Management 6](#_Toc328413440)

[2 General Instructions 7](#_Toc328413441)

[2.1 Message Structure 7](#_Toc328413442)

[2.2 Taxonomy and Mig Structure 7](#_Toc328413443)

[2.3 Schema Use 7](#_Toc328413444)

[3 Business Overview 8](#_Toc328413445)

[3.1 Financial Year and Substituted accounting periods. 8](#_Toc328413446)

[3.2 Schema Use by Date 8](#_Toc328413447)

[4 XBRL Context Specifications 9](#_Toc328413448)

[4.1 Context Specification Dimension 1: ReportPartyTypeDimension 9](#_Toc328413449)

[4.1.1 Context instances 10](#_Toc328413450)

[4.2 Context Specification Dimension 1: ReportPartyTypeDimension, Dimension 2: AssessableForeignIncomeClassificationDimension 11](#_Toc328413451)

[4.2.1 Context instances 12](#_Toc328413452)

[5 Interaction Model 13](#_Toc328413453)

[5.1 MESSAGE SPECIFICATION 13](#_Toc328413454)

[5.2 DIS LODGE specificationS 13](#_Toc328413455)

[5.2.1 DIS LODGE Request - Message 13](#_Toc328413456)

[5.2.1.1 Discoverable Taxonomy Set References 13](#_Toc328413457)

[5.2.1.2 Standard Business Document Header Content 13](#_Toc328413458)

[5.2.1.3 Standard Business Document Body Content 14](#_Toc328413459)

[5.2.2 DIS LODGE Response - Message 19](#_Toc328413460)

[5.2.2.1 Discoverable Taxonomy Set References 19](#_Toc328413461)

[5.2.2.2 Standard Business Document Header Content 19](#_Toc328413462)

[5.2.2.3 Standard Business Document Body Content 19](#_Toc328413463)

[Appendix A – The Message Content Table Explained 20](#_Toc328413464)

[Appendix B – Tax Office Structured English 23](#_Toc328413465)

[Appendix C – Validation Rules Alias Definitions 26](#_Toc328413466)

[Appendix D - Common Validation Rules 29](#_Toc328413467)

[Appendix E – Domain Definitions 35](#_Toc328413468)

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>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 Dividends and Interest Schedule (DIS) reporting taxonomy for the *Australian Taxation Office (ATO).*

The DIS cannot be sent in alone. It must accompany a Company Return (CTR). Therefore, refer to the Company Return (CTR) ‘parent’ MIG for details on relevant service.

* 1. Audience and Scope

This document contains the necessary information required to support the ATO DIS 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> | 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> | 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> | Reference information for software developers using the SBR software developer kit |

* 1. Change Management

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

1. General Instructions

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

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. DIS) with the return.

* 1. Message Structure

The message design of the ATO Income Tax suite specifies that the return and associated schedules (as required), are 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 detail 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. Business Overview

**Dividend and Interest schedule (DIS)**

Companies should lodge a Dividend and Interest schedule showing:

* the names, addresses, dates of birth, gender and tax file numbers (TFNs) or Australian business numbers (ABNs) (where quoted) of all shareholders (including employee shareholders in a consolidated or MEC group) to whom dividends (or deemed dividends) have been paid during the income year, including the amount of dividend paid to each shareholder and any franking credits for that amount.
* the names, addresses, dates of birth, gender and TFNs or ABNs (where quoted) of all investors, other than those investors in the business of providing business or consumer finance, to whom interest of $1 or more was paid or credited during the income year, and the amount of interest paid or credited to each person.

For further information about DIS, please refer to the company tax return instructions. This can be found on the ATO website: <http://ato.gov.au/taxprofessionals/content.asp?doc=/content/00189344.htm>

* 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 different 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

In SBR the Dividend and Interest Schedule will have a new reporting taxonomy released each year. An expiry date will not be specified as the reporting taxonomy will continue to be valid for that income year in the future. For example, in 2010 the DIS.0001 reporting taxonomy will be published, and in 2011 the DIS.0002 reporting taxonomy may be published.

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

This MIG corresponds to SBR reporting taxonomy dis.0001.02.00.

1. XBRL Context Specifications

The following sections define the context specifications 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 Imp** | **SBR Msg 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.GEN.000240 | 1. CMN.ATO.GEN.200007 |
| Entity Identifier | Mandatory | This field must be set to the TFN of the entity that the part of the business document instance relates to.  1.IF entity.identifier = NULLORBLANK  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.GEN.001020 | 1. CMN.ATO.GEN.001020 |
| Entity Identifier Scheme | Mandatory | This field must be set to <http://www.ato.gov.au/tfn>  1. IF (Identifier Scheme <> “<http://www.ato.gov.au/tfn>”)  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.GEN.001021 | 1. CMN.ATO.GEN.001021 |
| Entity Segment | Mandatory | Explicit member dimension ReportPartyType set to ReportingParty.  IF (RprtPyType.xx.xx:ReportPartyTypeDimension <> “RprtPyType.02.03:ReportingParty”)   RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.GEN.001019 | 1.CMN.ATO.GEN.001019 |
| Period Date - Start Date | Mandatory | 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 Date – End Date | Mandatory | 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 constrained values** | **Hypercubes** | **Instructions/Rules** | **Rule Imp** | **SBR Msg code** |
| --- | --- | --- | --- | --- | --- |
| **ReportPartyTypeDimension** |
| RP | “RprtPyType.02.03:ReportingParty” | ReportingParty | 1. IF (RP:entity.identifier.TFN <> PARENT RETURN:RP:entity.identifier.TFN)  RETURN VALIDATION MESSAGE ENDIF  2. IF COUNT(RP) <> 1  RETURN VALIDATION MESSAGE ENDIF  3. IF (RP:entity.identifier.TFN = NULLORBLANK)  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = CMN.ATO.GEN.402009  2. Schematron ID = VR.ATO.GEN.000208  3. Schematron ID = VR.ATO.GEN.410113 | 1. CMN.ATO.GEN.402009  2. CMN.ATO.GEN.200007  3. CMN.ATO.GEN.410113 |

* 1. Context Specification Dimension 1: ReportPartyTypeDimension, Dimension 2: AssessableForeignIncomeClassificationDimension

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| XBRL Instance Context Data Concept | Requirement | Instructions/Rules | Rule Imp | SBR Msg 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.GEN.000240 | 1. CMN.ATO.GEN.200007 |
| Entity Identifier | Mandatory | This field must be set to the TFN of the entity that the part of the business document instance relates to.  1. IF entity.identifier = NULLORBLANK  RETURN VALIDATION MESSAGE  ENDIF | 1. Schematron ID = VR.ATO.GEN.001020 | 1. CMN.ATO.GEN.001020 |
| Entity Identifier Scheme | Mandatory | This field must be set to <http://www.ato.gov.au/tfn>  1. IF (Identifier Scheme <> <http://www.ato.gov.au/tfn>)  RETURN VALIDATION MESSAGE  ENDIF | 1. Schematron ID = VR.ATO.GEN.001021 | 1. CMN.ATO.GEN.001021 |
| Entity Segment | Mandatory | Explicit member dimension ReportPartyType set to ReportingParty.  IF (RprtPyType.xx.xx:ReportPartyTypeDimension <> “RprtPyType.xx.xx:ReportingParty”)  RETURN VALIDATION MESSAGE  ENDIF | 1. Schematron ID = VR.ATO.GEN.001019 | 1.CMN.ATO.GEN.001019 |
|  | Mandatory | IF (RP:RprtPyType.xx.xx:AssessableForeignIncomeClassificationDimension <> SET(“AssblFrgnIncClass.02.00.ConduitForeignIncome”,” AssblFrgnIncClass.02.00.NonConduitForeignIncome”))  RETURN VALIDATION MESSAGE  ENDIF | 1. Schematron ID = VR.ATO.DIS.406000 | 1.CMN.ATO.DIS.406000 |
| Period Date - Start Date | Mandatory | 1. IF period.startDate = NULLORBLANK WHERE CONTEXT(ALL)  RETURN VALIDATION MESSAGE  ENDIF | 1. Schematron ID = VR.ATO.GEN.000199 | 1. CMN.ATO.GEN.001001 |
| Period Date - End Date | Mandatory | 1. IF period.endDate = NULLORBLANK WHERE CONTEXT(ALL)  RETURN VALIDATION MESSAGE  ENDIF | 1. Schematron ID = VR.ATO.GEN.000237 | 1.CMN.ATO.GEN.001001 |
| Scenario | Not Required | N/A | N/A | N/A |

* + 1. Context instances

| **Context instance MIG Label** | **Dimensions with constrained values** | | **Hypercubes** | **Instructions/Rules** | **Rule Imp** | **SBR Msg code** |
| --- | --- | --- | --- | --- | --- | --- |
| **ReportPartyType** | **AssessableForeignIncomeClassificationDimension** |
| RP.CFI | “RprtPyType.xx.xx:ReportingParty” | “AssblFrgnIncClass.02.00.ConduitForeignIncome” | AssessableForeignIncomeClassification | 1. IF (COUNT(RP.CFI) > 1)  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.DIS.406024 | 1.CMN.ATO.GEN.428278 |
| RP.NONCFI | “RprtPyType.xx.xx:ReportingParty” | “AssblFrgnIncClass.02.00.NonConduitForeignIncome” | AssessableForeignIncomeClassification | 1. IF (COUNT(RP.NONCFI) > 1)  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.DIS.406025 | 1.CMN.ATO.GEN.428278 |

# Interaction Model

* 1. MESSAGE SPECIFICATION

DIS will only be accepted by the ATO in a message with a CTR. Please refer to the MIG for the parent company return to determine message details.

* 1. DIS LODGE specificationS
     1. DIS LODGE Request - Message
        1. Discoverable Taxonomy Set References

|  |  |
| --- | --- |
| **Schema** | dis.0001.lodge.request.02.00.report.xsd  dis.0001.private.02.00.module.xsd |
| **Linkbases** | dis.0001.lodge.request.02.00.defLink.xml  dis.0001.private.02.00.defLink.xml |
|  | dis.0001.lodge.request.02.00.labLinkInfoCls.xml  dis.0001.private.02.00.labLinkInfoCls.xml |
|  | dis.0001.lodge.request.02.00.presLink.xml |
|  | dis.0001.lodge.request.02.00.refLink.xml  dis.0001.private.02.00.refLink.xml |
| **Example Instance** | dis.0001.lodge.request.02.00.sample.instance1.xml |
| **Schematron** | To be advised |

* + - 1. Standard Business Document Header Content

Refer to the ‘parent’ ITR MIG

* + - 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.

##### DIS LODGE Request XBRL Context

##### Refer to Section 4 XBRL CONTEXT SPECIFICATIONS

##### DIS LODGE Request Message Content Table

The following table contains the facts required in the instance document. Validation rule aliases used in these rules are fully defined in Appendix C – Validation Rules Alias Definitions. Aliases shown in these rules are represented as a word-number combination enclosed between square brackets, eg “[DISxxx]”.

###### DIS LODGE Request Message Content

| **Seq. No** | **Context** | **XBRL Fact** | **Instructions / Rules** | **Rule Imp** | **SBR Msg Code** |
| --- | --- | --- | --- | --- | --- |
| 1 | RP | 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 | 1. Schematron ID = VR.ATO.GEN.402010 | 1. CMN.ATO.GEN.402010 |
| **2** | **RP** | **orgname1.xx.xx:OrganisationNameDetails (Tuple 0..1)** | **N/A** | **N/A** | **N/A** |
| 2.1 | RP | pyde.xx.xx:OrganisationNameDetails.OrganisationalNameType.Code | 1. IF (pyde.xx.xx:OrganisationNameDetails.OrganisationalNameType.Code <> NULLORBLANK) AND (pyde.xx.xx:OrganisationNameDetails.OrganisationalNameType.Code <> "MN") RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.GEN.410049 | 1. CMN.ATO.GEN.410049 |
| 2.2 | RP | pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text | 1. IF ([DIS24] = NULLORBLANK) RETURN VALIDATION MESSAGE ENDIF  2. IF RP:pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text <> (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")  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.DIS.406005 2. Schematron ID = VR.ATO.GEN.400013 | 1. CMN.ATO.DIS.406005 2. CMN.ATO.GEN.400013 |
| **3** | RP | **prsnunstrcnm1.xx.xx:PersonUnstructuredName (Tuple 0..1)** | **N/A** | **N/A** | **N/A** |
| 3.1 | RP | pyde.xx.xx:PersonUnstructuredName.Usage.Code | 1. IF pyde.xx.xx:PersonUnstructuredName.Usage.Code <> NULLORBLANK AND pyde.xx.xx:PersonUnstructuredName.Usage.Code <> "Contact"  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.GEN.428209 | 1. CMN.ATO.GEN.428209 |
| 3.2 | RP | pyde.xx.xx:PersonUnstructuredName.FullName.Text | 1. IF ([DIS25] <> NULL) AND ([DIS25 > TEXT (200)) RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.DIS.406021 | 1. CMN.ATO.GEN.400011 |
| **4** | RP | **phone1.xx.xx:ElectronicContactTelephone (Tuple 0..1)** | **N/A** | **N/A** | **N/A** |
| 4.1 | RP | pyde.xx.xx:ElectronicContact.Telephone.Usage.Code | 1. IF ((pyde.xx.xx:ElectronicContact.Telephone.Usage.Code) <> NULLORBLANK) AND ((pyde.xx.xx:ElectronicContact.Telephone.Usage.Code) <> "03")  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.GEN.000051 | 1. CMN.ATO.GEN.000051 |
| 4.2 | RP | pyde.xx.xx:ElectronicContact.Telephone.ServiceLine.Code | N/A | N/A | N/A |
| 4.3 | RP | pyde.xx.xx:ElectronicContact.Telephone.Area.Code | N/A | N/A | N/A |
| 4.4 | RP | pyde.xx.xx:ElectronicContact.Telephone.Minimal.Number | 1. IF (pyde.xx.xx:ElectronicContact.Telephone.Minimal.Number = NULLORBLANK) AND (pyde.xx.xx:ElectronicContact.Telephone.Area.Code <> NULLORBLANK) RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.GEN.307006 | 1. CMN.ATO.GEN.307006 |
| **5** | RP | **dis.0001.lodge.req.xx.xx:Investor (Tuple 1..99)** | **N/A** | **N/A** | **N/A** |
| 5.1 | RP | 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 | 1. Agency | 1. CMN.ATO.GEN.428016 |
| 5.2 | RP | pyid.xx.xx:Identifiers.AustralianBusinessNumber.Identifier | 1. IF (pyid.xx.xx:IdentifiersAustralianBusinessNumberIdentifer <> NULLORBLANK) and (ABNALGORITHM(pyid.xx.xx:IdentifiersAustralianBusinessNumber.Identifer) = FALSE)  RETURN VALIDATION MESSAGE  ENDIF | 1. Schematron ID = VR.ATO.GEN.428040 | 1. CMN.ATO.GEN.000477 |
| 5.3 | RP | pyde.xx.xx:PersonDemographicDetails.Sex.Code | 1. IF (pyde.xx.xx:Person Demographic Details.Sex.Code <> NULLORBLANK AND pyde.xx.xx:Person Demographic Details.Sex.Code <> SET("1","2","3","0")) RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.DIS.406001 | 1. CMN.ATO.DIS.406001 |
| 5.4 | RP | pyde.xx.xx:PersonDemographicDetails.Birth.Date | 1. IF ([DIS13] > CURRENTDATE) RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.DIS.406020 | 1. CMN.ATO.DIS.406015 |
| 5.5 | RP | rvctc2.xx.xx:IncomeTax.FrankingCredits.ReceivedFromAustralianCompanies.Amount | 1. IF ([DIS16] > 0 AND [DIS17] = NULL) RETURN VALIDATION MESSAGE ENDIF  2. IF ([DIS17] > 2 AND [DIS16] < 1) RETURN VALIDATION MESSAGE ENDIF  3. IF ([DIS16] <= 1 AND [DIS17] <= 1 AND [DIS18] <= 1 AND [DIS19] <= 1 AND [DIS20] <= 1) RETURN VALIDATION MESSAGE ENDIF  4. IF ([DIS16] <> NULL) AND ([DIS16] <> MONETARY(U,12,0))  RETURN VALIDATION ERROR  ENDIF | 1. Schematron ID = VR.ATO.DIS.406003 2. Schematron ID = VR.ATO.DIS.406002 3. Schematron ID = VR.ATO.DIS.406004 4. Schematron ID = VR.ATO.DIS.406012 | 1. CMN.ATO.DIS.406003 2. CMN.ATO.DIS.406002 3. CMN.ATO.DIS.406004 4. CMN.ATO.GEN.400011 |
| 5.6 | RP | bafpo2.xx.xx:Equity.Dividends.FrankedPaid.Amount | 1. IF ([DIS17] <> NULL) and ([DIS17] <> MONETARY(U,12,0)) RETURN VALIDATION ERROR ENDIF | 1. Schematron ID = VR.ATO.DIS.406013 | 1. CMN.ATO.GEN.400011 |
| 5.7 | RP | bafpr2.xx.xx:Expense.Interest.Paid.Amount | 1. IF ([DIS20] <> NULL) and ([DIS20] <> MONETARY(U,12,0))  RETURN VALIDATION ERROR ENDIF | 1. Schematron ID = VR.ATO.DIS.406014 | 1. CMN.ATO.GEN.400011 |
| 5.8 | RP | pyde.xx.xx:Party.Type.Code | 1. IF ([DIS15] <>SET ("117","051","212","161","210","060") RETURN VALIDATION ERROR ENDIF 2. IF ([DIS15] <>SET ("117", "060") AND [DIS3] =NULLORBLANK) RETURN VALIDATION ERROR ENDIF 3. IF ([DIS15] = SET ("117", "060") AND [DIS4] = NULLORBLANK)  RETURN VALIDATION ERROR ENDIF | 1. Schematron ID = VR.ATO.DIS.406017 2. Schematron ID = VR.ATO.DIS.406015 3. Schematron ID = VR.ATO.DIS.406016 | 1. CMN.ATO.DIS.406014 2. CMN.ATO.DIS.406012 3. CMN.ATO.DIS.406013 |
| 5.9 | RP | pyde.xx.xx:PersonDemographicDetails.Deceased.Indicator | 1. IF ([DIS44] = TRUE AND [DIS15] <> “060”)  RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.DIS.406006 | 1. CMN.ATO.DIS.406006 |
| **5.10** | RP | **orgname1.xx.xx:OrganisationNameDetails (Tuple 0..1)** | **Standard Organisation Name rules apply. See APPENDIX D. Common Validation Rules** |  |  |
| 5.10.1 | RP | pyde.xx.xx:OrganisationNameDetails.OrganisationalNameType.Code | N/A | N/A | N/A |
| 5.10.2 | RP | pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text | 1. IF [DIS3] <> NULLORBLANK AND ([DIS4] <> NULLORBLANK OR [DIS5] <> NULLORBLANK OR [DIS6] <> NULLORBLANK)  RETURN VALIDATION MESSAGE  ENDIF | 1.Schematron ID = VR.ATO.DIS.406023 | 1.CMN.ATO.DIS.406023 |
| **5.11** | RP | **prsnstrcnm1.xx.xx:PersonNameDetails (Tuple 0..1)** | **Standard Person Name rules apply. See APPENDIX D. Common Validation Rules** | **N/A** | **N/A** |
| 5.11.1 | RP | pyde.xx.xx:PersonNameDetails.FamilyName.Text | N/A | N/A | N/A |
| 5.11.2 | RP | pyde.xx.xx:PersonNameDetails.GivenName.Text | N/A | N/A | N/A |
| 5.11.3 | RP | pyde.xx.xx:PersonNameDetails.OtherGivenName.Text | N/A | N/A | N/A |
| 5.11.4 | RP | pyde.xx.xx:PersonNameDetails.Position.Text | N/A | N/A | N/A |
| **5.12** | RP | **address1.xx.xx:AddressDetails (Tuple: 0..1)** | **Standard Address rules apply. See APPENDIX D: Common Validation Rules** |  |  |
| 5.12.1 | RP | pyde.xx.xx:AddressDetails.OverseasAddress.Indicator | 1. IF ([DIS41] = FALSE AND [DIS12] <> NULLORBLANK) RETURN VALIDATION MESSAGE ENDIF | 1. Schematron ID = VR.ATO.DIS.406010 | 1. CMN.ATO.DIS.406010 |
| 5.12.2 | RP | pyde.xx.xx:AddressDetails.Usage.Code | N/A | N/A | N/A |
| 5.12.3 | RP | pyde.xx.xx:AddressDetails.AttentionTo.Text | N/A | N/A | N/A |
| 5.12.4 | RP | pyde.xx.xx:AddressDetails.Line1.Text | 1. IF ([DIS7] = NULLORBLANK)  RETURN VALIDATION MESSAGE  ENDIF | 1. Schematron ID = VR.ATO.DIS.406022 | 1.CMN.ATO.DIS.406016 |
| 5.12.5 | RP | pyde.xx.xx:AddressDetails.Line2.Text | N/A | N/A | N/A |
| 5.12.6 | RP | pyde.xx.xx:AddressDetails.Line3.Text | N/A | N/A | N/A |
| 5.12.7 | RP | pyde.xx.xx:AddressDetails.Line4.Text | N/A | N/A | N/A |
| 5.12.8 | RP | pyde.xx.xx:AddressDetails.LocalityName.Text | N/A | N/A | N/A |
| 5.12.9 | RP | pyde.xx.xx:AddressDetails.Postcode.Text | N/A | N/A | N/A |
| 5.12.10 | RP | pyde.xx.xx:AddressDetails.StateOrTerritory.Code | N/A | N/A | N/A |
| 5.12.11 | RP | pyde.xx.xx:AddressDetails.CountryName.Text | N/A | N/A | N/A |
| 5.12.12 | RP | pyde.xx.xx:AddressDetails.Country.Code | N/A | N/A | N/A |
| Please note: to represent both Conduit and Non-Conduit Foreign Income Unfranked Dividend amounts, two instances of the UnfrankedDividends tuple (Seq No. 5.13 and 5.14 respectively) are required.  This tuple must be nested within the Investor tuple | | | | | |
| **5.13** | **RP.CFI** | **dis.0001.lodge.req.xx.xx:UnfrankedDividends (Tuple 0..1)** | **N/A** | **N/A** | **N/A** |
| 5.13.1 | RP.CFI | bafpo2.xx.xx:Equity.Dividends.UnfrankedPaid.Amount | 1. IF ([DIS19] <> NULL) AND ([DIS19] <> MONETARY(U,12,0))  RETURN VALIDATION ERROR  ENDIF | 1. Schematron ID = VR.ATO.DIS.406019 | 1. CMN.ATO.GEN.400011 |
| **5.14** | **RP.NONCFI** | **dis.0001.lodge.req.xx.xx:UnfrankedDividends (Tuple 0..1)** | **N/A** | **N/A** | **N/A** |
| 5.14.1 | RP.NONCFI | bafpo2.xx.xx:Equity.Dividends.UnfrankedPaid.Amount | 1. IF ([DIS18] <> NULL) AND ([DIS18] <> MONETARY(U,12,0))  RETURN VALIDATION ERROR ENDIF | 1. Schematron ID = VR.ATO.DIS.406018 | 1. CMN.ATO.GEN.400011 |

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

No XBRL instance will be returned.

* + - 1. Standard Business Document Header Content

Refer to the ‘parent’ ITR MIG

* + - 1. Standard Business Document Body Content

##### No XBRL instance will be returned for this schedule.

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 Instance which has 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.

| **Seq No.** | **XBRL Fact** |
| --- | --- |
|  | Payment Mechanism (tuple: zero to many) |
|  | Payment Mechanism.Payment Method.Code |
|  | Payment Mechanism.Instruction.Text |
|  | DirectDebit (tuple zero to one) |
|  | Payment Mechanism.Direct Debit Account.Identifier |
|  | Payment Mechanism.Direct Debit Name.Text |
|  | DirectCredit (tuple zero to one) |
|  | Payment Mechanism.Direct Credit Account.Identifier |
|  | Payment Mechanism.Direct Credit Name.Text |
|  | FinancialInstitutionAccount (tuple One Only) |
|  | Financial Institution Account.Bank State Branch.Number |
|  | Financial Institution Account.Financial Institution Account.Number |
|  | Financial Institution Account.Financial Institution Account Name.Text |
|  | Financial Institution Account.Financial Institution Branch Name.Text |
|  | Financial Institution Account.Financial Institution Name.Text |

**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 rules in this document are described in Tax Office Structured English. The intention is that the rules may be described unambiguously. Explanations of the terms used are detailed in the table below.

| **Structured English additional terms** | **Comments** |
| --- | --- |
| NULL | Fact is not there or has been specified to be null with xsi:nil indicator |
| NULLORBLANK | Fact is not there, is null with xsI:nil = true or is a null string "" |
| FORMAT |  |
| NUMERIC | Contains only digits between 0..9 |
| ASSOCIATED WITH |  |
| IN TUPLE | Usage: **IF <A> IN TUPLE <B>** |
| DIMENSION |  |
| WHERE |  |
| MONTH |  |
| YEAR |  |
| FINANCIAL YEAR |  |
| ANY CHARACTER OF | Any character within a field |
| <> SET ("0","1","2",",") | Set of acceptable characters |
| AND | Logical AND |
| OR | Logical OR |
| NOT | Reverses the value of a boolean. Ie turns TRUE to FALSE and vice versa |
| SET | Definition of an explicit set of values of which where one value meets the criteria for comparison a true response is given.  Usage: **SET("<b>","<c>","<d>")**  Note: No Spaces exist between the SET and (, Values are in brackets and are comma separated  Example Usage 1: **<a> <> SET("<b>","<c>","<d>")** is read as: <a> not any of <b>, <c> or <d> |
| DOMAIN | Set of a globally defined set of values    Example Usage **<a> = SET(DOMAIN(<B>))**  <a> is one of the values defined in <B> |
| CONTAINS | A string search for text within a field    Usage: **<a> CONTAINS <B>** |
| STARTSWITH | A string searches for text at the start of a field  Usage: **<a> STARTSWITH <B>** |
| ENDSWITH | A string searches for text at the end of a field  Usage: **<a> ENDSWITH <B>** |
| COUNT | A count of the number of occurrences of a field or context |
| - | Used to specify a range of values within the SET construct e.g. SET(0-9) |
| PARENT RETURN | Some schedule rules depend on the return it is lodged with.  Usage: **IF PARENT RETURN = “<A>”**  <A> could be CTR or PTR or other return |
| CONTEXT | Used to refer to a context instance  Usage: **CONTEXT(<A>)**  where <A> is a context instance abbreviation eg RP.GST.CC |
| SCHEDULE | To describe a schedule that could be lodged with a parent return.  Usage: **SCHEDULE = “<A>”**  Where <A> is a schedule abbreviation eg DIS, IEE |
| ALL OCCURENCES OF | For testing values in repeating tuples. |
| ANY OCCURRENCE OF | For testing values in repeating tuples. |
| ANY OTHER OCCURRENCE OF | For testing a value in one occurrence against other occurrences |
| SUM | The sum of all appearances of an element.  Usage: **SUM(<A>)**  where <A> is an element that appears in a repeating tuple or is a repeating element. |
| +/- | Usage: **A <> B +/- 1**  means(A > (B + 1)) OR (A < (B - 1))  Usage 2: **A = B +/- 1**  means (A <= (B + 1)) AND (A >= (B - 1)) |
| <IDtype>ALGORITHM | Usage: **IF TFNALGORITHM(<a>) = FALSE**  means the TFN field <a> fails the TFN algorithm  Usage: **IF ABNALGORITHM(<a>) = FALSE**  means the ABN field <a> fails the ABN algorithm  The algorithm are standard algorithms defined for each identifier by the Tax Office.  <IDtype> can be ABN, TFN, TAN, ARBN, ACN |
| MONETARY() | Definition of a valid monetary field pattern where a true response is given when a value passes all conditions.  Usage: **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  Examples: MONETARY(S,13,2)  MONETARY(U,13,2)  MONETARY(S,11,0)  Note: for <a> an S indicates a field can be prefixed with a sign, but it does not need to include one. However where <a> is a U the field cannot be prefixed with a sign. |
| NUMBER() | Definition of a valid numeric field pattern where a true response is given when a value passes all conditions.  Usage: **NUMBER(<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  Examples: NUMBER(S,13,2)  NUMBER(U,13,2)  NUMBER(S,11,0)  Note: for <a> an S indicates a field can be prefixed with a sign, but it does not need to include one. However where <a> is a U the field cannot be prefixed with a sign. |
| TEXT() | Definition of a valid text field pattern where a true response is given when a value passes all conditions.  Usage: **TEXT(<a>)**  Where <a> = Maximum number of characters  TRUE if the tested field is less than or equal to length <a> |
| COUNT(SCHEDULE = "aaa") = b | To describe if a schedule is or is not lodged with a parent return.  Usage: **COUNT(SCHEDULE = "<a>") = <b>**  Where <a> is a schedule abbreviation eg DIS, CGTS, S25A  <b> is count  Example: where a schedule is lodged with a parent[[1]](#footnote-1)  COUNT(SCHEDULE = "S25A") = 1  Example: where a schedule is lodged with a parent  COUNT(SCHEDULE = "S25A") = 0 |
| ABSVALUE | Absolute value. Ignore the signage. Both negative and positive values are included as positive values |

Appendix C – Validation Rules Alias Definitions

| **Alias** | **Definition** |
| --- | --- |
| DIS1 | DIS:RP:pyid.xx.xx:Identifiers.TaxFileNumber.Identifier IN TUPLE(dis.0001.lodge.req.xx.xx:Investor) |
| DIS2 | DIS:RP:pyid.xx.xx:Identifiers.AustralianBusinessNumber.Identifier IN TUPLE(dis.0001.lodge.req.xx.xx:Investor) |
| DIS3 | DIS:RP:pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text WHERE (pyde.xx.xx:OrganisationNameDetails.OrganisationalNameType.Code = "MN") IN TUPLE(orgname1.xx.xx:OrganisationNameDetails) IN TUPLE(dis.0001.lodge.req.xx.xx:Investor) |
| DIS4 | DIS:RP:pyde.xx.xx:PersonNameDetails.FamilyName.Text IN TUPLE(prsnstrcnm1.xx.xx:PersonNameDetails) IN TUPLE(dis.0001.lodge.req.xx.xx:Investor) |
| DIS5 | DIS:RP:pyde.xx.xx:PersonNameDetails.GivenName.Text IN TUPLE(prsnstrcnm1.xx.xx:PersonNameDetails) IN TUPLE(dis.0001.lodge.req.xx.xx:Investor) |
| DIS6 | DIS:RP:pyde.xx.xx:PersonNameDetails.OtherGivenName.Text IN TUPLE(prsnstrcnm1.xx.xx:PersonNameDetails) IN TUPLE(dis.0001.lodge.req.xx.xx:Investor) |
| DIS7 | DIS:RP:pyde.xx.xx:AddressDetails.Line1.Text WHERE (pyde.xx.xx:AddressDetails.Usage.Code = "POS") IN TUPLE(address1.xx.xx:AddressDetails) IN TUPLE(dis.0001.lodge.req.xx.xx:Investor) |
| DIS8 | DIS:RP:pyde.xx.xx:AddressDetails.Line2.Text WHERE (pyde.xx.xx:AddressDetails.Usage.Code = "POS") IN TUPLE(address1.xx.xx:AddressDetails) IN TUPLE(dis.0001.lodge.req.xx.xx:Investor) |
| DIS9 | DIS:RP:pyde.xx.xx:AddressDetails.LocalityName.Text WHERE (pyde.xx.xx:AddressDetails.Usage.Code = "POS") IN TUPLE(address1.xx.xx:AddressDetails) IN TUPLE(dis.0001.lodge.req.xx.xx:Investor) |
| DIS10 | DIS:RP:pyde.xx.xx:AddressDetails.StateOrTerritory.Code WHERE (pyde.xx.xx:AddressDetails.Usage.Code = "POS") IN TUPLE(address1.xx.xx:AddressDetails) IN TUPLE(dis.0001.lodge.req.xx.xx:Investor) |
| DIS11 | DIS:RP:pyde.xx.xx:AddressDetails.Postcode.Text WHERE (pyde.xx.xx:AddressDetails.Usage.Code = "POS") IN TUPLE(address1.xx.xx:AddressDetails) IN TUPLE(dis.0001.lodge.req.xx.xx:Investor) |
| DIS12 | DIS:RP:pyde.xx.xx:AddressDetails.CountryName.Text WHERE (pyde.xx.xx:AddressDetails.Usage.Code = "POS") IN TUPLE(address1.xx.xx:AddressDetails) IN TUPLE(dis.0001.lodge.req.xx.xx:Investor) |
| DIS13 | DIS:RP:pyde.xx.xx:PersonDemographicDetails.Birth.Date IN TUPLE(dis.0001.lodge.req.xx.xx:Investor) |
| DIS14 | DIS:RP:pyde.xx.xx:PersonDemographicDetails.Sex.Code IN TUPLE(dis.0001.lodge.req.xx.xx:Investor) |
| DIS15 | DIS:RP:pyde.xx.xx:Party.Type.Code IN TUPLE(dis.0001.lodge.req.xx.xx:Investor) |
| DIS16 | DIS:RP:rvctc2.xx.xx:IncomeTax.FrankingCredits.ReceivedFromAustralianCompanies.Amount IN TUPLE(dis.0001.lodge.req.xx.xx:Investor) |
| DIS17 | DIS:RP:bafpo2.xx.xx:Equity.Dividends.FrankedPaid.Amount IN TUPLE(dis.0001.lodge.req.xx.xx:Investor) |
| DIS18 | DIS:RP.NONCFI:bafpo2.xx.xx:Equity.Dividends.UnfrankedPaid.Amount IN TUPLE(dis.0001.lodge.req.xx.xx:UnfrankedDividends) IN TUPLE(dis.0001.lodge.req.xx.xx:Investor) |
| DIS19 | DIS:RP.CFI:bafpo2.xx.xx:Equity.Dividends.UnfrankedPaid.Amount IN TUPLE(dis.0001.lodge.req.xx.xx:UnfrankedDividends) IN TUPLE(dis.0001.lodge.req.xx.xx:Investor) |
| DIS20 | DIS:RP:bafpr2.xx.xx:Expense.Interest.Paid.Amount IN TUPLE(dis.0001.lodge.req.xx.xx:Investor) |
| DIS23 | DIS:RP:pyid.xx.xx:Identifiers.AustralianBusinessNumber.Identifier |
| DIS24 | DIS:RP:pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text WHERE (pyde.xx.xx:OrganisationNameDetails.OrganisationalNameType.Code = "MN") IN TUPLE(orgname1.xx.xx:OrganisationNameDetails) |
| DIS25 | DIS:RP:pyde.xx.xx:PersonUnstructuredName.FullName.Text WHERE (pyde.xx.xx:PersonNameDetails.Usage.Code = "Contact") IN TUPLE(prsnunstrcnm1.xx.xx:PersonUnstructuredName) |
| DIS26 | DIS:RP:pyde.xx.xx:ElectronicContact.Telephone.Area.Code WHERE (pyde.xx.xx:ElectronicContact.Telephone.Usage.Code = "03") IN TUPLE(phone1.xx.xx:ElectronicContactTelephone) |
| DIS27 | DIS:RP:pyde.xx.xx:ElectronicContact.Telephone.Minimal.Number WHERE (pyde.xx.xx:ElectronicContact.Telephone.Usage.Code = "03") IN TUPLE(phone1.xx.xx:ElectronicContactTelephone) |
| DIS32 | DIS:RP:pyde.xx.xx:ElectronicContact.Telephone.Usage.Code WHERE (pyde.xx.xx:ElectronicContact.Telephone.Usage.Code = "03") IN TUPLE(phone1.xx.xx:ElectronicContactTelephone) |
| DIS33 | DIS:RP:pyde.xx.xx:ElectronicContact.Telephone.ServiceLine.Code WHERE (pyde.xx.xx:ElectronicContact.Telephone.Usage.Code = "03") IN TUPLE(phone1.xx.xx:ElectronicContactTelephone) |
| DIS34 | DIS:RP:pyde.xx.xx:OrganisationNameDetails.OrganisationalNameType.Code WHERE (pyde.xx.xx:OrganisationNameDetails.OrganisationalNameType.Code = "MN") IN TUPLE(orgname1.xx.xx:OrganisationNameDetails) |
| DIS36 | DIS:RP:pyde.xx.xx:PersonUnstructuredName.Usage.Code WHERE (pyde.xx.xx:PersonNameDetails.Usage.Code = "Contact") IN TUPLE(prsnunstrcnm1.xx.xx:PersonUnstructuredName) |
| DIS37 | DIS:RP:pyde.xx.xx:AddressDetails.Usage.Code WHERE (pyde.xx.xx:AddressDetails.Usage.Code = "POS") IN TUPLE(address1.xx.xx:AddressDetails) IN TUPLE(dis.0001.lodge.req.xx.xx:Investor) |
| DIS38 | DIS:RP:pyde.xx.xx:AddressDetails.Line3.Text WHERE (pyde.xx.xx:AddressDetails.Usage.Code = "POS") IN TUPLE(address1.xx.xx:AddressDetails) IN TUPLE(dis.0001.lodge.req.xx.xx:Investor) |
| DIS39 | DIS:RP:pyde.xx.xx:AddressDetails.Line4.Text WHERE (pyde.xx.xx:AddressDetails.Usage.Code = "POS") IN TUPLE(address1.xx.xx:AddressDetails) IN TUPLE(dis.0001.lodge.req.xx.xx:Investor) |
| DIS40 | DIS:RP:pyde.xx.xx:AddressDetails.Country.Code WHERE (pyde.xx.xx:AddressDetails.Usage.Code = "POS") IN TUPLE(address1.xx.xx:AddressDetails) IN TUPLE(dis.0001.lodge.req.xx.xx:Investor) |
| DIS41 | DIS:RP:pyde.xx.xx:AddressDetails.OverseasAddress.Indicator WHERE (pyde.xx.xx:AddressDetails.Usage.Code = "POS") IN TUPLE(address1.xx.xx:AddressDetails) IN TUPLE(dis.0001.lodge.req.xx.xx:Investor) |
| DIS44 | DIS:RP:pyde.xx.xx:PersonDemographicDetails.Deceased.Indicator IN TUPLE(dis.0001.lodge.req.xx.xx:Investor) |
| DIS53 | DIS:RP:pyde.xx.xx:OrganisationNameDetails.OrganisationalNameType.Code WHERE (pyde.xx.xx:OrganisationNameDetails.OrganisationalNameType.Code = "MN") IN TUPLE(orgname1.xx.xx:OrganisationNameDetails) IN TUPLE(dis.0001.lodge.req.xx.xx:Investor) |
| DIS54 | DIS:RP:pyde.xx.xx:AddressDetails.AttentionTo.Text WHERE (pyde.xx.xx:AddressDetails.Usage.Code = "POS") IN TUPLE(address1.xx.xx:AddressDetails) IN TUPLE(dis.0001.lodge.req.xx.xx:Investor) |
| DIS55 | DIS:RP:pyde.xx.xx:PersonNameDetails.Position.Text IN TUPLE(prsnstrcnm1.xx.xx:PersonNameDetails) IN TUPLE(dis.0001.lodge.req.xx.xx:Investor) |

Appendix D - Common Validation Rules

**Common Address rules**

| **XBRL Fact** | **Common Address Rules** | **Rule Implementation** | **SBR Message Code** |
| --- | --- | --- | --- |
| AddressDetails.OverseasAddress.Indicator | IN TUPLE(address1.xx.xx:AddressDetails) IF (pyde.xx.xx:AddressDetails.OverseasAddress.Indicator = NULLORBLANK)  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.000500 | CMN.ATO.GEN.410167 |
| pyde.xx.xx:AddressDetails.Line1.Text | IN TUPLE(address1.xx.xx:AddressDetails) IF (pyde.xx.xx:AddressDetails.Line1.Text = NULLORBLANK)  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.000501 | CMN.ATO.GEN.410214 |
| pyde.xx.xx:AddressDetails.Country.Code | IN TUPLE(address1.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.000502 | CMN.ATO.GEN.410212 |
| pyde.xx.xx:AddressDetails.Country.Code | IN TUPLE(address1.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.000503 | CMN.ATO.GEN.410191 |
| pyde.xx.xx:AddressDetails.Country.Code | IN TUPLE(address1.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.000504 | 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.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 | IN TUPLE(address1.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.000506 | CMN.ATO.GEN.410002 |
| pyde.xx.xx:AddressDetails.Line1.Text | IN TUPLE(address1.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.000507 | CMN.ATO.GEN.410003 |
| pyde.xx.xx:AddressDetails.Line1.Text | IN TUPLE(address1.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.000508 | 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 | IN TUPLE(address1.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.000509 | 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 | IN TUPLE(address1.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.000510 | 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 | IN TUPLE(address1.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.000511 | CMN.ATO.GEN.410017 |
| pyde.xx.xx:AddressDetails.StateOrTerritory.Code | IN TUPLE(address1.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.000512 | CMN.ATO.GEN.410018 |
| pyde.xx.xx:AddressDetails.Usage.Code | IN TUPLE(address1.xx.xx:AddressDetails) IF (pyde.xx.xx:AddressDetails.Usage.Code = NULLORBLANK)   RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.000513 | CMN.ATO.GEN.410174 |

**Common Organisation Name rules**

| **XBRL Fact** | **Common Organisation Name Rules** | **Rule Implementation** | **SBR Message Code** |
| --- | --- | --- | --- |
| 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 | IN TUPLE(orgname1.xx.xx:OrganisationNameDetails) IF (pyde.xx.xx:OrganisationNameDetails.OrganisationalNameType.Code = NULLORBLANK)   RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.000520 | CMN.ATO.GEN.410147 |
| pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text | IN TUPLE(orgname1.xx.xx:OrganisationNameDetails) IF (pyde.xx.xx:OrganisationNameDetails.OrganisationalName.Text = NULLORBLANK)  RETURN VALIDATION MESSAGE ENDIF | Schematron ID = VR.ATO.GEN.000521 | CMN.ATO.GEN.410215 |

**Common Person Name rules**

| **XBRL Fact** | **Common Person Name Rules** | **Rule Implementation** | **SBR Message Code** |
| --- | --- | --- | --- |
| pyde.xx.xx:PersonNameDetails.FamilyName.Text | IN TUPLE(prsnstrcnm1.xx.xx:PersonNameDetails) IF (pyde.xx.xx:PersonNameDetails.FamilyName.Text = NULLORBLANK)   RETURN VALIDATION MESSAGE ENDIF | VR.ATO.GEN.000530 | 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.OtherGivenName.Text | IN TUPLE(prsnstrcnm1.xx.xx:PersonNameDetails)  IF (pyde.xx.xx:PersonNameDetails.GivenName.Text = NULLORBLANK)   RETURN VALIDATION MESSAGE ENDIF | VR.ATO.GEN.000531 | 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 |

Appendix E – Domain Definitions

**DOMAIN(COUNTRY CODES)**

The domain of valid Country codes is defined in the SBRAU 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 |

1. *Note: For a schedule of IEE, the valid expression is* ***COUNT(SCHEDULE = “IEE”) >= 1*** *as it is permissible to have more than 1 IEE schedule lodge with the relevant parent form.* [↑](#footnote-ref-1)