**SECTION A1  
Table of Contents**

1.1 Introduction 3

1.2 ELS Takeon Development and Support 3

1.3 Registration of ELS Software 3

1.4 (Not Allocated) 4

1.5 ELS Tax Agent Guide 4

1.6 How To Use These Specifications 5

1.7 Testing ELS Software In Preparation For Tax Office Registration 6

1.7.1 The Test Gateway 6

1.7.2 How to gain access to the Test Gateway. 6

1.8 Transmission Media 6

1.9 Transmission Structure 6

1.9.1 File Names 7

1.9.2 Combined transmissions 7

1.9.3 Transmission of returns for Prior Year (PY) 7

1.10 Data Structure 8

1.11 Schedules 8

1.12 ELS Validation 9

1.12.1 Operating system 9

1.13 Tax Agent’s Certificate/Electronic Lodgment Declaration 9

1.14 Key to Table Headings Appearing in Other Sections Within This Specification. 10

1.14.1 Tag Files – Sections B, C & D 10

1.14.2 Edits - Section E, Parts 1 and 2 11

1.14.3 Note References and Warnings Chapter 11

1.15 Rules Which Apply To All Forms and Schedules 12

1.15.1 Global Edits 12

1.15.2 Generic Edits 12

1.15.3 Data Types 13

1.15.4 Multi-Valued Fields 16

1.15.5 Associated Groups 17

1.15.6 Multi-valued associated groups 17

1.15.7 Maximum Length 18

1.15.8 Duplicate Files 18

1.15.9 Tags 19

1.15.10 Mandatory Fields 19

1.15.11 The Use of "NULL" 20

1.15.12 Zero Values 21

1.15.13 NULL To Zero Function 21

1.15.14 Automatic Insertion Of Zero 22

1.15.15 Changes From Set Defaults 22

1.15.16 SUM function 23

1.15.17 Dollar Tolerance 23

1.15.18 Cyclical edits containing “Year of return - ^ABB” 24

1.15.19 IF Statement 25

1.15.20 MOD statement 26

1.15.21 Action of edits with all relevant tags absent or null 26

1.15.22 Tag Order 26

1.16 Tables 27

1.16.1 Tax Office ANZSIC Codes (Table 2) 27

1.17 Paper Formats 27

## 1.1 Introduction

The Electronic Lodgment Service (ELS) provides an electronic means for Tax Agents to lodge tax returns and other forms to the Tax Office.

When sent to the Tax Office, these electronic forms must comply with a strict data format. Each data item contained in a form must comply with all applicable edit rules.

For every financial year, a completely new version of the ELS Software Specification is issued. This copy of the specifications refers only to the current year return forms, reports and other forms. For most form types, ELS will also accept lodgment of last year’s returns. The data on these returns must comply with the latest version of last year’s specifications. This new version may be subject to minor changes throughout the year.

ELS will also accept lodgment of most prior year forms. The data on there forms must comply with the latest version of last year’s specifications. For example to successfully lodge a 2006 Individual Income Tax return form the ELS software must conform to the most recently released specification for that tax year ie 2006-10.

The ELS also facilitates the delivery of a number of tax agent practice management reports and Activity Statements.

## 1.2 ELS Takeon Development and Support

These specifications are prepared by members of the Tax Time project team. The project team also administers the registration of ELS software, maintains test gateways and is responsible for the installation of the new version of the ELS validation system just prior to the commencement of the new financial year.

All queries should be lodged with SILU to ensure tracking of the query and correct escalation.

## 1.3 Registration of ELS Software

Each year ELS software must be registered with the tax office to allow for the transmission of data through the ELS. The registration process is undertaken to ensure that each form produced by a software house meets the minimum edit and data standards as set out in the ELS specifications. This is achieved by demonstrating that forms can be successfully transmitted to the Tax Office by passing through the ELS Validation system.

The Tax Time project team administers the registration and defines the testing process to be undertaken.

Changes to the specifications can be made at any time throughout the year. Although no further testing or further registration requirements are imposed, it is usually mandatory to make corresponding changes to ELS software where applicable.

ELS Software must also comply with the Document Interchange Specifications -version 11.28. These can be downloaded from the software developers web site http://softwaredevelopers.ato.gov.au/.

## 1.4 (Not Allocated)

## 1.5 ELS Tax Agent Guide

It is a **mandatory** requirement to supply the hyperlink to the Tax Office’s *Electronic Lodgment Service (ELS) guide for tax professionals* (also known as the ELS guide) on the ato.gov.au website to Tax Agents via your normal software distribution process.

The link is:

<http://www.ato.gov.au/Tax-professionals/Electronic-lodgment-service/In-detail/ELS-Guide/>

The ELS guide will be available in Word format to all software producers, through the software developers website, by the end of March.

It is your responsibility as a software developer to determine how best to include this hyperlink in your product for Tax Agents. Ideally, information such as that included in *Appendix A — Explanation of Return Form Errors* should also be integrated within the software package (for example, as a simple help file with a search capability).

Software developers should be aware of the following copyright provisions for the *Electronic Lodgment Service (ELS) guide for tax professionals*:

© Commonwealth of Australia 2011

This work is copyright. You may download, display, print and reproduce this material in unaltered form only (retaining this notice) for your personal, non-commercial use or use within your organisation. Apart from any use as permitted under the Copyright Act 1968, all other rights are reserved. Requests for further authorisation should be directed to the Commonwealth Copyright Administration, Intellectual Property Branch, Department of Communications, Information Technology and the Arts, GPO Box 2154, Canberra ACT 2601 or posted at www.dcita.gov.au/cca

## 1.6 How To Use These Specifications

In producing ELS software you must read and apply the information provided in the relevant sections of these ELS Specifications. (Note that the section letters are represented in the document names).

**Section A, Part 1:** Contains information which is referred to throughout other chapters and explains their meaning. To produce any form it is essential you read all of this.

**Section A, Part 2:** Specifies the business rules or any specific supplementary **i**nformation which may assist you in the design of each form.

**Section A, Part 3:** Provides an overview of reports sent to Tax Agents through ELS. This includes information about Transmission and Validation reports.

**Section A, Part 4:** Contains algorithms required to verify certain numbers used by the ATO

**Section A, Part 5:** Specifies the requirements of ELS Declarations.

**Section A, Part 6:** This document describes the Document Interchange System (DIS\_SBR) via SBR and the requirements for software communicating with DIS\_SBR.

**Section B:** Is generally referred to as the ‘Tag Chapters’. The documents which make up this section specify each field on each form. It lists the edits and notes which apply to each. Each document covers a different return form.

**Section C:** Specifies each field on each schedule. Each document covers a different schedule.

**Section D:** Specifies each field relating to each report. Each document covers a different report.

**Section E, Part 1:** Specifies each edit, including all those referred to in the ‘Tag chapters’ and each Global Edit. The edits within these documents cover all return forms, reports and schedules. This document also displays edit messages as they appear on validation reports and edit explanations as provided in the ELS Guide.

**Section E, Part 2:** Specifies each function referred to by the edits with the Edit Test starting "Check" and provides further explanation of edits and functions.

**Section E, Part 3:** Describes notes which apply to each field and warning messages.

**Section F:** Lists each table referred to in Edits and Functions.

**Section G:** Contains a number of appendices, including an alphabetical list of tags, test numbers and checklist items.

## 1.7 Testing ELS Software In Preparation For Tax Office Registration

The Tax Time project prepares a test package for each form type contained in these specifications. These packages, usually referred to as ‘Cross Check cases’, are generally available within 6 weeks after the specification is finalised. This test package refers to the procedures required to obtain Tax Office registration for each form. For most form types a new registration is required each year.

### 1.7.1 The Test Gateway

The Tax Office maintains a Test Gateway for software producers to test their software. This gateway allows returns to be transmitted via modem and to be validated by the Tax Office ELS validation system. Almost immediate feedback on how well the software follows the applicable ELS specifications is available through the validation report.

The Test Gateway also contains a Bulletin Board which contains a copy of all current ELS system specifications.

### 1.7.2 How to gain access to the Test Gateway.

The Tax Office will register your request and supply you with an approval number which you can use to log into the Test Gateway. When you first log in your login ID and password will be the same. You can alter your password to any combination of twelve or more alphabetic or numeric characters, provided it meets password requirements as specified by the current DIS specification.

## 1.8 Transmission Media

The Electronic Lodgment Service (ELS) receives electronic forms in the following way:-

via the Internet to the ELS Document Interchange System (ELS/DIS). Modem transmissions must follow the communications protocol as specified in the separate ELS Document Interchange System Version 11.28(B) specifications

**Note:**  Floppy diskettes are no longer an acceptable format for transmission of files to the Tax Office.

## 1.9 Transmission Structure

A transmission is a collection of data files made up of one Transmission Identification file (TXID)-see Section A, Part 2 (*A02\_form.doc*), - followed by one or more return files. Each return or form must be in a separate file. Up to 1000 return files can be included in a single transmission. A single transmission can be:-

transmitted via the Internet, during a single communications session. A transmission occurs each time an agent starts a file transmission protocol (Kermit or ZModem) in the transmit mode.

### 1.9.1 File Names

Each file within a transmission must be given a filename. Within a single transmission each filename must be unique.

Each transmission must include one, and only one, Transmission Identification File (TXID) file. The TXID file must always have the filename 'TXID'.

All other filenames are limited to 16 characters. The first character must be an upper case alphabetic character. The other characters may be any numeric or upper case alphabetic character, the full stop or the underscore character.

Only the following characters may be used in a valid filename:

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

1 2 3 4 5 6 7 8 9 0 \_ .

Filenames may conform to MS DOS (or PC DOS) filename conventions, including the use of 'dot' extensions.

These filenames are used as a form of identification in reports provided to Tax Agents.

NOTE: Filenames cannot commence with 'TRANS', 'VALID' or 'BULL'.

### 1.9.2 Combined transmissions

A transmission containing more than one type of return is called a combined transmission. All the form types specified in this document, with the exception of the Superannuation Surcharge and Client Preference (CP) forms, can be combined in a single transmission.

### 1.9.3 Transmission of returns for Prior Year (PY)

ELS can accept current year returns, or those from prior years back to and including 1998. PY returns and schedules must comply with the ELS Specifications which applied in the prior year. These PY returns must be accompanied by a TXID file complying with the prior year specifications when there are only prior year files present. Where there is a combination of PY files and current year files, the TXID file must comply with current year specifications (ie. those in the document ‘*B01\_txid.doc’*)

## 1.10 Data Structure

ELS uses a ‘Tag’ file structure. Each field or data item on a form is represented by a 'tag'. Tags are made up of 4 characters, commencing with the caret character (for example ^AED or ^ATY). The data immediately following a tag pertains to that tag. Each data item is terminated by the next tag, which can relate to another data item or be the tag ^ZZZ, which indicates the end of the file.

Each file must be assembled into a continuous sequence of characters comprising tags and data. Files must not contain carriage return or line feed characters.

This is an example of a TXID file:

^AAW21092001^AAT138111^AAS12345^AARUMG^AAZP^AAM1.0^AAN13^ZZZ

This example can be interpreted as:

**Tag Data Item Description Data (or Value)**

^AAW Submission Date 21 September 2001

^AAT Software Registration No. 138111

^AAS Transmitter Registration No. 12345

^AAR Office of Lodgment UMG

^AAZ Transmission Status P

^AAM Software Version 1.0

^AAN Number of Returns Transmitted 13

^ZZZ (Signifies end of file)

Tags can generally be sent in any order, with four exceptions, in which the order is important. The exceptions are:-

(a) Associated Groups (see 1.15.5 below);

(b) Multi-Valued Associated Groups (see 1.15.6 below);

(c) Schedules (see 1.11 below); and

(d) tag ^ZZZ, which must appear at the end of each file.

## 1.11 Schedules

Schedules are a form which cannot be sent in as a file on its own, but must be 'attached' to a form. A form must precede all schedules. More than one schedule can be 'attached' to a form, unless precluded by an edit.

Each schedule starts with an identifying tag made up of the characters ‘^XXX’, followed by a single character identifying the schedule type. For example, ^XXXT indicates the start of data for a Missing Group Certificate Schedule.

All data belonging to a schedule must be within the return file. Schedule data must end either with the tag identifying another schedule, or the end-of-file tag (^ZZZ).

If a schedule identification tag is transmitted, then the mandatory data items and those items required by edits, must also be present. If, for example, a return file contained the tag ^XXXE immediately followed by the end-of-file tag ^ZZZ, the return will be rejected by the ELS validation program as the required data items would be missing.

## 1.12 ELS Validation

All ELS data transmitted to the Tax Office goes through the ELS Validation system which checks that the data is acceptable. Any files containing data not conforming to the ELS Software Specifications will be rejected by the Tax Office and require correction by the Tax Agent prior to its acceptance by the Tax Office. To negate the need for Tax Agents to re-key and re-transmit their data, ELS Software must subject all data to all applicable edit tests, as specified in this document, prior to transmission.

The status or outcome of each transmission is provided to the transmitting agency through two reports, the Transmission Report and the Validation Report. The document ‘*A03\_reps.doc*’ (Section A, Part 3) provides further information about these reports.

### 1.12.1 Operating system

The ELS Validation System is an application residing on the Tax Office CEG. It is a Unix operating system using Universe BASIC and Universe databases.

The edit tests described in these specifications are the same as those used by programmers within the tax office to build the ELS validation system.

## 1.13 Tax Agent’s Certificate/Electronic Lodgment Declaration

Tax Agents who lodge any of the forms listed in Section A part 5 using ELS must ensure a declaration is completed. The Employer or Taxpayer must sign a declaration. If Electronic Funds Transfer (EFT) is requested, an EFT Declaration must also be signed by the taxpayer.

Form specific declarations can be found at Section A part 5.

ELS Software must include a facility to print these declarations. When printed, fields such as Tax File Number, Phone number, etc, may be derived directly from the return data. Alternatively these fields may be left blank and be hand written by the Tax Agent or taxpayer.

Declaration forms produced by ELS Software must contain exactly the same wording, follow the same sequence, contain the same items and follow the same general format as the forms shown in Section A part 5. The software must ensure that users will be able to easily ascertain which declaration needs to be printed, (ie. in reference to the form type which has been completed).

## 1.14 Key to Table Headings Appearing in Other Sections Within This Specification.

### 1.14.1 Tag Files – Sections B, C & D

The column headings of the tag file sections (all documents prefixed with ‘B’, ’C’ or ‘D’ in the filename), are defined as follows:

|  |  |
| --- | --- |
| **Field** | **Description** |
| **Tag** | A three character alphabetical code preceded by the ^ character, which generally represents a field on the return form or schedule. |
| **M** | Mandatory status. If a Y is present then the field represented by tag is mandatory. Global edit V17 will apply. |
| **Data Type** | The data type pertaining to that field. (See Data Types, below, for a full description). Global edits apply. |
| **MV** | Multi-valued. |
| **Assoc Group** | Indicates the fields which form part of the associated group including this field. The tag group is displayed, eg. ^AQA - ^AEF. |
| **Description** | Description of the field as it appears on the paper return form. |
| **Item** | Numeric reference as appears on the paper return. |
| **Label** | Alphabetical reference as appears on the paper return. |
| **Max Length** | The maximum number of characters allowed for a field. Global edit V7 applies. |
| **Edit Ref** | Edit reference. Lists all the edit tests applicable to that field, excluding global edits (see 1.12.1, below). Section E provides full details about each edit. |
| **Note Ref** | Note reference. All the note references and warning messages applicable to a field. Notes provide supplementary information related to a field. Warning messages (if mandatory) must appear in software when the specified condition occurs. Section E, Part 3 (*E3\_note.doc*) provides full details about each note and warning message. |

### 1.14.2 Edits - Section E, Parts 1 and 2

All edits are fully specified in Section E, Part 1, (the document *‘E1\_edit.doc’)*, except for those where the Edit Test begins with the word ‘Check’. These ‘check edits’, referred to as ‘Generic’ edits, are further specified in Part 2 (*E2\_edit.doc*), (see 1.12.2, below). Part 2 also contains a range of additional information about edits, including definitions of operants, and a list of global edits.

All edits are written in the fail condition, ie. if the condition specified by an edit is true, then the data is invalid.

The column headings of the Edits Section E, Part 1, are defined as follows:

|  |  |
| --- | --- |
| **Field** | **Description** |
| **Edit Ref** | Edit reference. This is the alphanumeric reference of the edit test. Software must report this reference when a test fails. |
| **M** | Mandatory status. If a 'Y' is present the edit test is mandatory. If blank, the edit does not apply to ELS software and is for Tax Office use only. |
| **Edit Test** | Description of the edit test usually expressed in structured English. These are written in the fail condition. When preceded by the word ‘Check’, a function is called. Functions are specified Part 2 of the Edits section (E2\_edit.doc). |
| **Message** | This is the message which is reported on Validation Reports provided to Tax Agents when the edit test fails. A similar message must also be displayed in ELS software if the condition occurs. Software may expand or adjust this message. |
| **Explanation** | The edit explanation provides a detailed description of why the edit triggered. It is provided to the Tax Agents via the ELS Guide. This message may be included in ELS software if the condition occurs. |
| **Form Type** | The Form Type(s) and Schedule(s) to which that edit applies. |

### 1.14.3 Note References and Warnings Chapter

Notes provide some elaboration or further information related to a field. Warnings exist to counter probable input errors. Notes and warnings are specified in Section E, Part 3 (*E3\_note.doc*). Notes are prefixed ‘N’; warnings with ‘M’.

Some notes refer to tables. If a note is mandatory and a table has been referred to, then the table must form part of the software.

Warning messages, if mandatory (ie. the M column is marked ‘Y’) must be implemented in the software. Each reference defines the condition and message that is to be displayed. A warning message must prompt the keying operator, wherever the message condition is met, to initiate corrective action, if required. They carry informative status only and are not fatal. Warning messages are not coded in the Tax Office ELS validation.

The column headings of this section, are defined as follows:

|  |  |
| --- | --- |
| **Field** | **Description** |
| **Note Ref** | Reference for the notes or warnings as appears in the ‘Tag’ chapters. Notes are prefixed by ‘N’. Warning messages are prefixed by ‘M’. |
| **M** | Mandatory. A 'Y' signifies that the note or warning must be incorporated in software. |
| **Description** | A description of the requirements for each note or warning |

## 1.15 Rules Which Apply To All Forms and Schedules

### 1.15.1 Global Edits

Global edits are edits which *are not specifically referred to in the ‘Tag’ chapters*, but must be implemented. They apply to all fields.

All global edits are mandatory.

A typical example of a global edit is V7, (maximum field length exceeded). Although not specifically listed against any field in the ‘Tag’ chapters, V7 applies to all fields with a field length specified.

All 'global edits' are fully specified in Section E, Part 1 (ie*. E1\_edit.doc)*, and are listed separately in Section E, Part 2, (*E2\_edit.doc*).

### 1.15.2 Generic Edits

Generic edits are preceded by the word ‘Check’, [eg. V210F: CheckAbsentState(Tag1,Tag2)]. Rather than specifying a specific tag within the edit, tags are listed as ‘Tag1, Tag2, Tag3, etc. The details on which tags actually apply and how the edit operates is provided in Section E, Part 2 (*E2\_edit.doc*).

### 1.15.3 Data Types

For each field listed in the ‘Tag’ chapters a ‘Data Type’ is specified. The following details specify each data type.

| **Field** | **Description** |
| --- | --- |
| **Date8** | Must be represented as eight numerics in the format DDMMCCYY where DD is the day, MM is the month and CCYY is a four character representation which allows the full year to be shown. For example, ‘23101896’ is 23rd October 1896.  *Global edit V11 applies when a field of this data type has the incorrect format.* |
| **$** | Whole dollar money fields, positive values only. Must only contain numerics. They must not contain /L, decimal point, spaces, cents, dollar sign, cent sign, minus sign, plus sign, text or fractions of a dollar. For example, $38.00 must be transmitted as 38.  *Global edit V23 applies when a field of this data type has the incorrect format.* |
| **$/L** | Whole dollar money fields, positive and negative values are acceptable. Fields of this data type can be followed by an optional ‘/L’, used to indicate a negative value. They must not contain decimal point, spaces, cents, dollar sign, cent sign, brackets, plus sign, minus sign, text or fractions of a dollar. As an example, $38 Loss must be transmitted as ‘38/L’.  *Global edit V40 applies when a field of this data type has the incorrect format.* |
| **$c** | Dollar and cent money fields, positive values only. For example, $380,000.50 must be shown as 380000.50. If there are no cents the amount is to be shown as 380000.00. Digits to the left of the decimal point are interpreted as whole dollars, and those to the right as cents. A fractional amount of less than $1, eg. 50 cents, must be shown with a leading zero in the format '0.50'. Fractions of a cent must be truncated to the nearest cent. This data type must not contain /L, spaces, dollar sign, cent sign, brackets, plus sign, minus sign, or text characters. As valid examples, $38.10c must be shown as 38.10; and $38.00c as 38.00.  *Global edit V38 applies when a field of this data type has the incorrect format.* |
| **$c/L** | Dollar and cent money fields, positive and negative values are acceptable. Fields of this type can be followed by an optional ‘/L’, used to indicate a negative value. A decimal point must be included. Digits to the left of the decimal point are interpreted as whole dollars, and those to the right as cents. A fractional amount of less than $1, eg 50 cents, must be shown with the leading zero in the format ‘0.50’. Fractions of a cent must be truncated to the nearest cent. This data type must not contain spaces, dollar sign, cent sign, brackets, plus sign, minus sign, or text characters. As valid examples, $38.10c must be shown as 38.10, $38.00 must be shown as 38.00, a loss of $23.45c must be shown as 23.45/L, and a loss of $23.00 must be shown as 23.00/L.  *Global edit V2231 applies when a field of this data type has the incorrect format.* |
| **Integer** | Integer fields can only contain whole numbers.  *Global edit V44 applies when a field of this data type has the incorrect format.* |
| **Question** | Fields of data type Question must be Y, N or ‘null’. If a field with a data type of question field is mandatory, Y or N must be transmitted.  *Global edit V43 applies when a field of this data type has the incorrect format.* |
| **Text** | Text fields may contain any of the following characters:  A B C D E F G H I J K L M N O P Q R S T U V W X Y Z a b c d e f g h i j k l m n o p q r s t u v w x y z 1 2 3 4 5 6 7 8 9 0 ! @ $ % & ( ) - = [ ] ; : ' " . ? / space \* ,  ie. the following chars are not valid: # ` { } | \ ^ + ~ < > \_  *Global edit V9 applies when a field of this data type has the incorrect format* |
| **Name Text** | Name text fields may contain any of the following characters, (note-the differences with 'Text' are shown after ‘space’ on both data types):  A B C D E F G H I J K L M N O P Q R S T U V W X Y Z a b c d e f g h i j k l m n o p q r s t u v w x y z 1 2 3 4 5 6 7 8 9 0 ! @ $ % & ( ) - = [ ] ; : ' " . ? / space ` # \_ { } | \  ie. the following chars are not valid: , ^ \* + ~ < >  *Global edit V42 applies when a field of this data type has the incorrect format.* |
| **Restricted Name Text** | Restricted name text fields may contain any of the following characters: A B C D E F G H I J K L M N O P Q R S T U V W X Y Z a b c d e f g h I j k l m n o p q r s t u v w x y z 1 2 3 4 5 6 7 8 9 0 ' – space  ie. the following chars are not valid: , ^ \* + ~ < > ! @ $ % & ( ) = [ ] ; : " . ? / ` # \_ { } | \  *Global edit V15 applies when a field of this data type has the incorrect format.* |
| **Address Text** | Address text fields may contain any of the following characters A B C D E F G H I J K L M N O P Q R S T U V W X Y Z a b c d e f g h i j k l m n o p q r s t u v w x y z 1 2 3 4 5 6 7 8 9 0 ' - space & / ( )  ie. the following chars are not valid: , ^ \* + ~ < > ! @ $ % = [ ] ; : " . ? ` # \_ { } | \  *Global edit V71 applies when a field of this data type has the incorrect format.* |
| **Time4** | Must be all numerics and in the format HHMM where HH is hours and MM is minutes, therefore, the last 2 numerics must not be greater than 59.  *Global edit V12 applies when a field of this data type has the incorrect format.* |
| **Dec2** | Must be all numerics and in the format nnnnnnnn.nn (i.e.meaning the actual length of the number). The characters to the left of the decimal place are integers and the characters to the right of the decimal place are a fraction of an integer (expressed in hundredths).  *Global edit V13 applies when a field of this data type has the incorrect format.* |
| **Dec3** | Must be all numerics and in the format n.nnn (i.e.meaning the actual length of the number). The characters to the left of the decimal place are integers and the characters to the right of the decimal place are a fraction of an integer (expressed in thousands).  *Global edit V14 applies when a field of this data type has the incorrect format.* |
| **Dec4** | Must be all numerics and in the format nnnn.nnnn (i.e.meaning the actual length of the number). The characters to the left of the decimal place are integers and the characters to the right of the decimal place are a fraction of an integer (expressed in ten thousands).  *Global edit V37 applies when a field of this data type has the incorrect format.* |
| **Dec5** | Must be all numerics and in the format n.nnnnn (i.e meaning the actual length of the number). The characters to the left of the decimal place are integers and the characters to the right of the decimal place are a fraction of an integer (expressed in 100 thousands).  *Global edit V31 applies when a field of this data type has the incorrect format.* |
| **Free Text** | Any character, except ^ and ~.  No specific global edit applies when ^ or ~ is transmitted. However, because the caret ‘^’ character signifies a tag, edit V3 is likely. The tilde char ‘~’ is a special character which will be read incorrectly by the Tax Office’s UNIX system.   Software must ensure these characters cannot be transmitted as data. |

### 1.15.4 Multi-Valued Fields

Some data fields may be present more than once. These fields are called multi-valued fields. When transmitted, each value of the field must be preceded by its tag.

Multi-valued fields in the tag descriptions for all forms and schedules are represented by a column headed ‘MV’. If a tag is multi-valued, a text character is inserted in the column headed MV.

A number in the MV column indicates the maximum number of times that a field can be transmitted, and the character M (for many) in this field indicates any number of occurrences.

Where there is no value in the MV column this indicates only one occurrence is acceptable.

### 1.15.5 Associated Groups

Some data fields are related to other data fields. These fields are called associated fields and all the fields related to each other are defined as an associated group.

Associated groups are identified in the specifications by an entry in the column titled ‘Assoc Group’. The first and last tags in the group, as they appear in the specifications, are listed. For example on the 2004 Individual Form (EI), Group Certificates are represented by the associated group ^GFA-^AEF, which includes the tags ^GFA, ^AQB, ^AQE, ^AEI, ^AED,^HDZ, ^AEJ and ^AEF, in that order.

When any value is entered against a tag which belongs to an associated group, all tags within that group must be included in the transmission (output file). Even if one or more of the tag values are NULL, all tags must still be sent. (If all tag values are NULL, then none of the tags should be sent). The tags must be sent in the order defined in this specification. Tags which do not form part of the associated group cannot be sent within the group – they must come before or after tags belonging to the associated group.

Global Edit V5 will occur if tags are missing or if the specified sequence is not followed.

### 1.15.6 Multi-valued associated groups

Multi-valued associated groups contain both a value in the ‘MV’ column and in the 'Assoc Group’ column within the specifications. They are groups which may occur more than once within a transmission. The rules applying to the transmission of Associated Groups, as defined above, continue to apply, with the additional requirement that each group is separated from the other(s).

This may be best illustrated with an example:-

On the 2004 Individual Form, the tags ^GFA, ^AQB, ^AQE, ^AEI, ^AED,^HDZ, ^AEJ and ^AEF are an associated group which cover the information required from a Group Certificate. ‘^GFA-^AEF’ is written in the column named ‘Assoc Group’ for each of these tags. ‘M’ is written in the field marked ‘MV’, meaning that each tag and group can appear many times. In other words, a taxpayer may have many Group Certificates. If there were two Group Certificates, all the data for the first certificate would be sent, followed by all the data for the second.

Two examples of valid transmissions of multi-valued associated groups containing Group Certificate information follows (the return characters are included to aid readability).

^GFA84111122223

^AQB2300.00

^AQE11000

^AEI

^AED

^HDZ

^AEJ

^AEF

^GFA98111133334

^AQB2600.00

^AQE12000

^AEI5.00

^AED160

^HDZ

^AEJ25.00

^AEF1600

Global Edit V5 will occur if tags are missing or if the specified sequence is not followed.

### 1.15.7 Maximum Length

The maximum field length for each field is specified in the ‘Max Length’ column in Sections B, C and D. If blank, no limit is set. Data fields must not contain leading or trailing spaces or zeros, except when entering a dollar and cent money field that has a fractional amount less than $1, eg. 0.50

Global Edit V7 will occur if a maximum field length is exceeded.

NOTE: As there is only 1 length attribute associated $/L type tags this means that in effect if the maximum length is 12, then V7 permits a 12 digit positive number but only a 10 digit loss.

### 1.15.8 Duplicate Files

The Tax Office validation system retains a table containing the Tax File Numbers (TFNs) and return form types as well as Document Identification Numbers (DINs) of accepted and processed ELS transmissions. If a Tax Agent sends in a form with the same TFN and return form type, or same DIN as one held on this table, a validation edit will be reported to the Tax Agent and the form will be rejected. This condition does *not* apply to Forms CP, CU, EM, SP, SR, ES, EX, EY and RR

To reduce the occurrence of errors of this type, software must include a facility which checks that a form for a particular client has not been sent previously to the Tax Office, while providing an override facility to allow a form to be re-sent if required. This override facility should have a higher access level to prevent common usage.

### 1.15.9 Tags

There are three kinds of tag:

**(i) Tags Representing Fields**

All fields or data items are represented by a tag. These have the format ^AAA, where ^ is the ASCII character with the decimal value 94 and AAA are three alphabetic characters. Within a transmission, all data is preceded by a tag.

**(ii) Tags Representing Schedules**

The data derived from a schedule is transmitted with return form data. To signify that data is derived from a schedule a tag in the format ^XXXAA precedes a schedule (where AA is a single or double alphabetic character representing the type of schedule). Egs. ^XXXA is Schedule A and ^XXXBA is Schedule BA.

**(iii) End of File Tag**

At the end of each file, the end-of-file tag ‘^ZZZ’ must be transmitted. This tag must not be followed by any other characters in the return file.

### 1.15.10 Mandatory Fields

Some data fields are designated mandatory. Mandatory fields must be present and not equal to null. Such fields are indicated in the ELS Software Specifications in Sections B, C & D, (ie. in the *tag* sections) by a Y character in the M column. Mandatory fields must be transmitted with a valid value.

A mandatory field must have a tag, for example (^AAA) followed by at least one character that is not a space. If the value of a mandatory money field type is zero, the value 0 or 0.00 (depending on the data type) must be transmitted.

A tag with no value (ie. = NULL), is treated by the Tax Office as if it were a 'blank space' on paper. The Tax Office does not assign any value (even zero) to such a field.

Mandatory fields in a schedule must be present only when that schedule is present. Eg. if Schedule A is required, ^AEB must be transmitted with a value.

To reduce size and complexity, non-mandatory fields must be omitted when they have no value, unless they belong to an associated group and one or more fields in the group has a value.

Global Edit V17 will occur if a mandatory tag is missing.

### 1.15.11 The Use of "NULL"

Throughout this Specification, "NULL" is defined as:

1. No value after the tag

Examples:

'^AAA^AAB1' where ^AAA is null

or

^AAA

^AAB1

Where AAA is null and is followed immediately by the end of the line.

2. No tag is present

Examples:

'^AAB1' where ^AAA is not present;

or

^AAA12

^AAC13

Where ^AAB is not present.

If a value, including zero, is required to satisfy an edit, then 'null' is not valid.

A tag with no value is treated by the Tax Office as if it were a blank space on paper. The Tax Office does not assign any value (even zero) to such a field.

Where required to include a null tag in an associated group (as described under Associated Groups above) include the tag as in the first definition of null above.

### 1.15.12 Zero Values

Where the value of zero is shown on screen or printed on the paper return, it must be transmitted electronically. A zero must only be shown on screen and transmitted when it is required to satisfy an edit. Null used as a value for zero is not acceptable.

Data fields must not contain leading or trailing spaces or zeros. The only exception to this rule is the use of a leading zero when entering a dollar and cent money field that has a fractional amount less than $1 (eg. 0.50).

### 1.15.13 NULL To Zero Function

NULL.TO.ZERO() is a function within the Takeon Validation System that is used in edits to convert null values to zero. The function is applied to a tag and if the value of the tag is null, zero is returned, otherwise the tag value is returned unchanged.

***Why do we use NULL.TO.ZERO()?***

The NULL.TO.ZERO() function is used because of two conditions in UniVerse Basic:

(1) Calculations involving only null values produces a zero value

(2) Null is less than zero

In most edits involving calculations, it is implied that we do not want to differentiate between null and zero – that is, we wish to treat null as if it were zero.

For example:

^FGA < ^BYB + ^BYD + ^AWK

In this case, a return is rejected if the sum of the 3 tags ^BYB, ^BYD and ^AWK is greater than the tag ^FGA. However, because none of the above tags are mandatory, the scenario that all 4 tags are null is quite possible.

Note that the edit implies that we only perform the edit test if at least one of the tags involved in the edit has a value. That is to say, it is as if the edit were written:

(^FGA < ^BYB + ^BYD + ^AWK) and (^FGA # NULL or ^BYB # NULL or ^BYD # NULL or ^AWK # NULL)

If all 4 tags are null:

^BYB + ^BYD + ^AWK = 0 (according to condition (1))

^FYG < 0 is true (according to condition (2))

Therefore the return will be rejected – *which is not the desired outcome*.

By applying the NULL.TO.ZERO() function to ^FYG in the edit code will circumvent this situation, and correct the problem. That is:

NULL.TO.ZERO(^FYG) < 0 is false. The return will not be rejected on the V1005 edit test.

***In which situations do we apply NULL.TO.ZERO()?***

In general, NULL.TO.ZERO() is applied to a tag in the code of edits following the format below:

1. tag < calculation
2. tag = calculation
3. tag # calculation
4. tag >= calculation

(where the calculation involves one or more tags)

We do *not* apply the NULL.TO.ZERO() function if the edits are in any of the following formats:

* tag > calculation
* tag <= calculation

because changing a null tag value to zero will not alter the value of the expression. NULL.TO.ZERO() is redundant in this case.

### 1.15.14 Automatic Insertion Of Zero

A zero cannot be automatically system inserted into a field unless an edit requires a value at that field.

Where a zero (0) is automatically inserted in a field to satisfy an edit, for any income field, the warning message "M511: Check that zero income is correct at 'field name'," must be displayed. (Examples of income fields are 'Taxable income of spouse', and 'Gross rent').

Where a zero is automatically inserted in a field to satisfy an edit at any non-income field, (eg. deductions, rebates, tax credit), the warning message, (M511) is not required.

The automatic insertion of a zero in the primary production income fields, to satisfy an edit, is not permitted, except where the taxpayer is a primary producer with primary production income.

### 1.15.15 Changes From Set Defaults

ELS fields or data items, including those in the TXID, may be used as defaults or master items within ELS software products.

If users change a set default or master item, either intentionally or by chance, this may cause problems with the successful transmission or processing of forms if the correct values are not used.

To prevent these types of errors, ELS software products should make use of an alert message to the user advising them to check and reset these values, or an alternative method tailored to suit your software application.

### 1.15.16 SUM function

To ensure the consistent approach to specifying the SUM{} function the following function names have been used.

New function names:

1. SUM.TAG{} – for multi-valued tags, this means sum of all of the values of the tag
2. SUM.SCHED{} – for schedules that can have multiple occurrences, this means sum of all of the occurrences of the (single value) tag across all of the schedules

**Examples:**

^ABC – multivalue tag coming in on 1 schedule AB

If ^ABC = 1, 2, 3 then:

SUM.TAG{^ABC} = 1+2+3 = 6

^DEF – non-multivalue tag coming in on 3 schedule ABs

If ^DEF = 10; ^DEF = 20; ^DEF = 30 then:

SUM.SCHED{^DEF} = 10+20+30 = 60

^GHI – multivalue tag coming in on 3 schedule ABs

If ^GHI = 1,2,3; ^GHI = 1,2,3; ^GHI = 1,2,3 then:

SUM.SCHED{SUM.TAG{^GHI}} = (1+2+3) + (1+2+3) + (1+2+3) = 18

 We cannot use just SUM.SCHED{^GHI} in this situation as ^GHI is multivalue.

**General Rules:**

1. Multi-value tag, 1 Form/Schedule – SUM.TAG{^tag}
2. Single-value tag, multiple schedules – SUM.SCHED{^tag}
3. Multi-value tag, multiple schedules – SUM.SCHED{SUM.TAG{^tag}

### 1.15.17 Dollar Tolerance

In cases where a field is an integer, but a calculation may **not** result in a whole number a dollar tolerance must be included to pass validation.

In more recent times these tolerances have been specified in edit tests, however some older edits may not openly described this tolerenace; in these cases it should be inferred that a + 1 or -1 tolerance is in place

### 1.15.18 Cyclical edits containing “Year of return - ^ABB”

From 2010 onwards, all of the “cyclical” type edits referencing a year will be rewritten to remove excess changes needed to the system. Rather than referencing the actual a year in an edit, for instance 2006, and then having to update it every year, ^ABB (year of return) will be used in the edit test where it is in the list of tags for that form.

Writing edits in this format means the edit test no longer requires updating each year.

The date reference is written in the following ways:

* Where the year is the same as the year of the return, DDMM:(^ABB). For a 2010 return, 30 June 2010 would be 3006:(^ABB).
* Where the year is different to the year of the return, DDMM:(^ABB-NN). For a 2010 return, 1 February 1995 would be 0102:(^ABB-15).

The “ **:** ” in the edit test is a format inclusion to succinctly separate the DDMM with the YYYY section of the date.

**Example 1**

**V154** (with reference to the 2009 year)

From

|  |  |
| --- | --- |
| Test | ^ABQ < **01061933** and Sum.Tag{^AMP} > 0 |
| Mess | The taxpayer has passed the eligible age where they can claim a deduction at D12 |
| Expl | When the taxpayer is over the age of 76, they are not entitled to a deduction for Personal superannuation contributions (Item D12 Label H). |

To

|  |  |
| --- | --- |
| Test | ^ABQ < **(‘0106’:(^ABB-76))** and Sum.Tag{^AMP} > 0 |
| Mess | The taxpayer has passed the eligible age where they can claim a deduction at D12 |
| Expl | When the taxpayer is over the age of 76, they are not entitled to a deduction for Personal superannuation contributions (Item D12 Label H). |

**Example 2**

**V1542** (with reference to the 2009 year)

From

|  |  |
| --- | --- |
| Test | ^ARA = **01072008** and ^ARB = **30062009** |
| Mess | Dates do not indicate a substituted accounting period |
| Expl | The substituted accounting period must be different to the usual financial year accounting period |

To

|  |  |
| --- | --- |
| Test | ^ARA = **(‘0107’:(^ABB-1))** and ^ARB = **(‘3006’:(^ABB))** |
| Mess | Dates do not indicate a substituted accounting period |
| Expl | The substituted accounting period must be different to the usual financial year accounting period |

### 1.15.19 IF Statement

In a number of edits, a tag is only included in the calculation that forms part of that edit when it meets a particular condition. This is written in the following format:

IF ^AAA condition ^AAA

**Example 1**

^ATY + (IF ^GAX # 8 ^GAY) >= 250000

In this example, the value at ^GAY would only be included if ^GAX does not equal 8

Assume the tags have the following values:

^ATY 200000

^GAX 8

^GAY 70000

The value at ^GAY is not included in the calculation as ^AGX is equal to 8. Therefore the edit will not report as ^ATY is less than 250000, even though the sum of ^ATY and ^GAY is 270000.

**Example 2**

(IF ^FLO > 3737 ^FLO) + ^JDZ + ^JEB >= 250000

In this example, the value at ^FLO will only be included if it is greater than 3738.

Assume the tags have the following values:

^FLO 8000

^JDZ 200000

^JEB 45000

The value at ^FLO is included in the calculation as it is greater than 3737. Therefore the edit will report as the sum of the three tags is 253000.

### 1.15.20 Action of edits with all relevant tags absent or null

Generic edits are always run against any return to which they are attached.

Other edits are not run against a return where no tags involved in that edit are present in the return.

**Example**

The edit *Any occurrence of (^JGL # 'I' and ^BBK = Null)* appears as if it would trigger where ^JGL and ^BBK are both null, however the edit would not trigger as it would not be run.

### 1.15.21 Tag Order

Data should be transmitted in the specified tag order for the form, as set out in the documents in sections B and C of this specification.

The first three tags of a return must be included in the correct order or certain global edits will not function as expected.

**Example**

V145 tests whether a form type (^AWR) and year or version (^ABB or ^BGF) is included in table 6.

If a return is transmitted as:

*^ABB2011^AWREI^AAE0013000*

V145 will trigger, treating the first tag (^ABB) as form type and the second (^AWR) as year/version.

## 1.16 Tables

Some Edit Tests and Note References refer to specific tables. These tables appear in Section F. Where a table is specified, it must be implemented and any edit tests or notes applicable to that table, must be applied.

### 1.16.1 Tax Office ANZSIC Codes (Table 2)

The Industry Codes, as listed in Table 2, are used for validation purposes only. The table contains a list of the hierarchical valid codes not a list of valid industries. Therefore, the table is not suitable for use as a look up table by Tax Agents in allocating a correct Tax Office ANZSIC code against a business activity description.

Look up tables should, if at all possible, be incorporated in software using data files provided by the Tax Office. The full list of Business Industry codes (Nat 1827) is available on [www.ato.gov.au](http://www.ato.gov.au).

## 1.17 Paper Formats

If software is to allow tax agents the option of lodging a paper version of a form, then the paper output must follow the Paper Facsimile format. Software packages must be granted a separate registration number from the Tax Office. The registration process is managed by SILU.