IPND Manager

Integrated Public Number Database - IPND Error Correction

Date: July 2019

Approved by: Tony Parker
Title: IPND Manager

TABLE OF CONTENTS

1. ...PURPOSE 3
2. ...SCOPE 3
3. ...ACTIONS 3
3.1. File Level Errors 3
3.2. Mandatory Hard Errors 30
3.3. Mandatory Soft Errors 45
3.4. Warnings 61
3.5. Obsolete 64
3.6. Process for IPND File Processing 66
3.6.1. File Upload 66
3.6.2. Error Processing - General Procedure 66
3.6.3. Validation of upload file 66
3.6.4. Data Record Processing 67
4. ...MEASURES 68
5. ...REFERENCES 688
6. ...DEFINITIONS 68
7. ...ATTACHMENTS 68
7.1. Error List 69
DOCUMENT CONTROL SHEET 72

1. PURPOSE

The purpose of this document is to describe the processes for the management and correction of errors resulting from erroneous or incomplete files being sent to the IPND.

2. SCOPE

This document will only cover the business processes regarding error management and correction in the IPND. Technical aspects of error management will not be considered in this document.

3. ACTIONS

This section describes individual errors and the context in which the error codes are generated in the IPND.

Each error falls into one of five categories:

- F File level errors - Section 3.1
- MH Mandatory Hard - Section 3.2
- MS Mandatory Soft - Section 3.3
- W Warning - Section 3.4
- * Obsolete - Section 3.5

Each error in each category will be addressed in three parts.

Background - will provide general information on the concepts covered and terms used and the objective of the error being issued.

Reason - will provide a synopsis of the error issued, explaining it's main aspects.

Corrective Action - will outline the action required to be taken to correct the error.

Also described is the business process for managing and correcting these errors - Section 3.6.

3.1. File Level Errors

File level errors are a method of checking and validating Upload Files sent to the IPND. They are a way by which the database confirms that a file has been transferred intact, without corruption and in the right sequence. The purpose of capturing errors at file level is essentially to guarantee the integrity and completeness of the file.

Errors resulting from the attempted loading of an upload file which has inconsistencies, at a file level, will result in an error code being written to the error file. This will indicate that the upload file was rejected and no records were processed in the IPND. As validation at the file level will be done prior to checking at the record or field level, no records will be processed until all the errors at file level have been corrected and the file has been resent.

If a file is rejected because of an error at file level for example the source ID code is invalid, no further files may be processed from that provider until a file arrives with that specific Sequence Number. Therefore if file 0000016 is rejected on day 1, file 0000017 that arrives on day 2 and 0000018 that arrives on day 3 will not be processed until file 0000016 is corrected and re-transmitted. Files 0000017 and 0000018 will also need to be re-transmitted after file 0000016 has been successfully processed. This rule is valid for all the error codes at file level. Irrespective of which error code is returned on a file, files sent after the file which has errored will not be processed . They will need to be re-transmitted after the erroneous file has been corrected, resent and processed.

A number of errors returned on the one file may not require each error to receive separate attention. For example:

IPNDPXXXXX.0000001

The above filename string will error as the U is missing in the file type field. Error 201 which refers to the incorrect upload filename length is not the only error that will be returned due to the incorrect filename example above. The way in which the database reads the records means that it will read the first six characters in the filename searching for the file type. This would result in IPNDPX being read as the file type, XXXX0 being read as the File Source code etc. What can be seen from this example is that the one missing character will result in several errors being returned. It is possible that only the one omission or invalid entry needs attention, to clear a number of errors. Correcting this one missing character will result in several errors being cleared. The above string would be corrected in the following way:

IPNDUPXXXXX.0000001

3.1.1. 001 - File out of sequence

Background

The IPND Upload Files provided by a specific File Source are uniquely identified by a Sequence Number. This number refers to the position of an individual file in relation to the other files provided from this File Source. The File Sequence number is a component of the filename, (it is also included in the header and trailer records for the file). The File Sequence number is therefore a count of the number of successfully processed Upload Files from a specific Data Provider’s source system to the IPND.

Explanation of Error
This error is issued when the sequence number for the Upload File due to be processed is not equal to the Sequence Number of the last successfully processed Upload File from the same File Source + 1. The File Sequence number of the last successfully processed Upload file will be compared with the File Sequence Number of the current file. If the current file sequence number does not equal the last file sequence number +1 the file will be rejected for an invalid sequence number and reported in the error file. Examples of files sent out of sequence follow:

For the purpose of this example XXXXX is a valid Data Provider File Source code.

Scenario 1

IPNDUPXXXXX.0000001	- 1st file sent by XXXXX - Accepted
IPNDUPXXXXX.0000002	- 2nd file sent by XXXXX- Accepted
IPNDUPXXXXX.0000002	- 3rd file out of sequence - Rejected

Scenario 2

IPNDUPXXXXX.0000001	- 1st file sent by XXXXX - Accepted
IPNDUPXXXXX.0000002	- 2nd file sent by XXXXX- Accepted
IPNDUPXXXXX.0000004	- 3rd file out of sequence - Rejected

Corrective Action
This error is corrected by finding out the sequence number of the last successfully uploaded file, usually uploaded the previous day and adding 1 to this figure.

Scenario 1

IPNDUPXXXXX.0000001	- 1st file sent by XXXXX - Accepted
IPNDUPXXXXX.0000002	- 2nd file sent by XXXXX- Accepted
IPNDUPXXXXX.0000002	- 3rd file out of sequence - Rejected
IPNDUPXXXXX.0000003	- 3rd file resent is in sequence - Accepted

Scenario 2

IPNDUPXXXXX.0000001	- 1st file sent by XXXXX - Accepted
IPNDUPXXXXX.0000002	- 2nd file sent by XXXXX- Accepted
IPNDUPXXXXX.0000004	- 3rd file out of sequence - Rejected
IPNDUPXXXXX.0000003	- 3rd file sent by XXXXX - Accepted

3.1.2. 018 - Incorrect File Source Code

Background
The File Source Code is a 5 character code embedded in the filename which identifies the system sending the data. Each Data Provider will have at least one File Source Codebut may have a number of different codes making it easily recognisable which Data Provider's source system has sent any particular file to the IPND. The File Source code is compared to a database of valid codes to determine the validity of the code. There will be no validation by the to ensure that the Data Provider code on a public number record is appropriate for the File Source code in the filename. The onus is on the Data Provider to ensure the

correct File Source code is used in the file, that the File Source code actually belongs to the Data Provider sending the file and that it reflects the source system that has sent the file. Validation of a File Source code against the Data Provider code cannot occur due to the fact that a Data Provider may have multiple systems each with it's own File Source code and each generating it's own sequence of files.

Explanation of Error
This error is issued when the File Source code that is provided in an Upload File is checked against the database of valid codes and found to be invalid.

Example - Data Provider A

IPNDUPXXXXX.0000001	- Valid code - Accepted
IPNDUPYYYYY.0000001	- Valid code - Accepted
IPNDUPXXXXY.0000001	- Invalid code - Rejected
IPNDUPZZZZZ.0000001	- Invalid code - Rejected

In the above example Data Provider A has two source systems which provide data to the IPND. The File Source code of one is XXXXX and the code of the other is YYYYY. If Data Provider A tried to send a File Source code of XXXXY or ZZZZZ or some other code not recognised by the database this error would be returned.

Corrective Action
This error is corrected by the file being retransmitted with a correct and appropriate File Source code. The File Source code should be valid and should reflect which source system has sent the file. If the source system supplying the data records has the File Source code XXXXX, YYYYY though it is a valid code recognised by the IPND, is not an appropriate code to use in this filename as YYYYY does not reflect the source system of the file. A file with a File Source code of YYYYY though would not be rejected as it is valid. This means that this particular file source code exists in the database table. The onus is on the Data Provider to ensure the File Source code is valid and appropriate, reflecting the source system which has generated the file.

If the source system supplying the file is reflected by File Source code XXXX the correct filename would be:

IPNDUPXXXXX.0000001

3.1.3. 201 - Incorrect upload filename length

Background
IPND Upload Files must all have a filename provided as part of the upload. This filename is a 19 character string detailing important information about the file.

A correct upload filename will have the following structure:

IPND<TT><XXXXX>.<NNNNNNN>

The 'IPND' is a constant in all file types.

<TT> refers to the file type, which in this case is 'UP' for upload file

<XXXXX> is the file source code particular to a source system of that Data Provider sending the file

'.' Is an extension marker

<NNNNNNN> is a file sequence number with leading zeros

Assuming that XXXXX is a valid Data Provider File Source code and this is the first file being sent an example of a valid filename would be:

IPNDUPXXXXX.0000001

Explanation of Error
This error is issued when the length of the file name is incorrect with either digits or characters missing or extra ones added in. As mentioned in the File Level errors introduction, the addition or omission of characters can affect the filename and header length which in turn affects the way the database will read the supplied information. Any files that fail to comply with the above naming structure will be rejected and reported in the error file. It is important to ensure that the filename is the correct length with no extra digits, characters or spaces and with no missing characters or spaces and that all the characters are in the correct position.

Examples

IPNDUPXXXXXX.0000001	- File Source code too long - Rejected
IPNDUPYYYY.0000001	- File Source code too short - Rejected
IPNDUPXXXXX..0000001	- Two separators provided - Rejected

Corrective Action
This error is corrected by ensuring the filename length is 19 characters and that each of these spaces is filled by an appropriate character. The correction of the above examples follows:

IPNDUPXXXXX.0000001	- Correct File Source code used - Accepted
IPNDUPYYYYY.0000001	- Correct File Source code used - Accepted
IPNDUPXXXXX.0000001	- Correct Sequence Number used - Accepted

3.1.4. 202 - Incorrect upload filename prefix

Background
The upload filename prefix is the first component of the filename. In the case of upload files to the IPND the filename prefix is "IPNDUP". The prefix is a method of identifying what type of file the IPND is receiving. This prefix indicates that the file being received by the IPND is a Data Provider upload file.

Explanation of Error
This error is issued when the upload file does not start with these 6 characters "IPNDUP". These 6 characters must immediately precede the file source code as part of the Upload File.

Examples of filenames that would error due to an incorrect filename are given below:

IP DUPXXXXX.0000001	- Blank space in filename prefix
IPNUPXXXXX.0000001	- Missing filename prefix character
IPNDU XXXXX.0000001	- Blank space in filename prefix

Corrective Action
This error is corrected by the provision of a correct and complete six character filename prefix. An example of a filename with a correct prefix that would enable the file to be uploaded appears below:

IPNDUPXXXXX.0000001

3.1.5. 203 - Invalid upload filename separator

Background
The filename separator is used to separate the File Source code from the File Sequence number. It is denoted by a full stop and appears as part of the upload header details as “.”.

Explanation of Error
If the upload file name does not have a “.” (the separator) in the twelfth position this error is generated. Examples where this error would be issued appear below:

IPNDUPXXXXX0000001	- No filename separator provided
IPNDUPXXXXX .0000001	- Filename separator not in the 12th position

Corrective Action
Upload Files must contain a separator represented as a full stop in the twelfth position in the filename field. The above two examples of filenames that would generate this error would be corrected by the example that appears below:

IPNDUPXXXXX.0000001

3.1.6. 204 - Incorrect upload filename sequence format

Background
The filename provides an effective way for the database to verify important details sent in the Upload file. In order to confirm these important details that have been sent the various components of the filename must be in the correct format and positioning.

A correct upload filename will have the following format:

IPND<TT><XXXXX>.<NNNNNNN>

<NNNNNNN> is the file sequence number with leading zeros

The file sequence number is a 7 digit field denoting the number of the upload file. For the very first file sent to the IPND the sequence number will be 0000001.

Explanation of Error
This error is returned when the file sequence number in the filename has been provided in an incorrect format. Any files that fail to comply with the above numbering format will be rejected and reported in the error file. The following examples show instances when an Upload File will be rejected.

IPNDUPXXXXX.000001	- Digit missing
IPNDUPXXXXX. 0000001	- Extra space included in sequence number
IPNDUPXXXXX.1	- Leading zeros missing

Corrective Action
This error is corrected when the file is retransmitted with a correctly formatted sequence number as part of the filename. An example of a filename with a correctly formatted sequence number follows:

IPNDUPXXXXX.0000001

3.1.7. 205 - Upload filename sequence does not match footer sequence

Background
As the transmission of a file over a live network has certain risks it is important to validate the integrity of any Upload Files received by the IPND and to confirm that the file has been transmitted in its entirety. The filename precedes all the data records sent as part of a file and the trailer record completes the file as it is appended to the file after all the data records have been included. One way of confirming that a file has been transmitted intact is to check that the filename sequence and the footer sequence in the Upload File match.

Explanation of Error
This error is returned when the upload filename sequence does not match the filename sequence entry in the trailer record of the file. For example:

File Name : IPNDUPXXXXX.0000001

Trailer : TRL00000121999040411234400000016

Corrective Action
In this example the actual file sequence number is 0000001 but the file sequence entry in the trailer is 0000012. For this error to be corrected the correct sequence number must be included in the trailer data record as shown below:

File Name : IPNDUPXXXXX.0000001

Trailer : TRL00000011999040411234400000016

3.1.8. 206 - Upload filename sequence does not match header sequence

Background
This error code was designed as a form of validation, performed by the database in order to ensure that the sequence numbers are the same both in the filename and the header record. This assists in confirming that the file has been transmitted uncorrupted and intact.

Explanation of Error
This error is returned when the upload filename sequence does not match the filename sequence entry in the header record of the file. For example:

File Name : IPNDUPXXXXX.0000001

Header : HDRIPNDUPXXXXX.00000121998042811223 3

Corrective Action
In this example the actual file sequence number is 0000001 but the file sequence entry in the header is 0000012. For this error to be corrected the correct sequence number must be included in the header data record as shown in the example below:

File Name : IPNDUPXXXXX.0000001

Header : HDRIPNDUPXXXXX.00000011998042811223 3

3.1.9. 207 - Invalid upload filename source code

Background
The filename source is a code used to uniquely identify the Data Provider's source system which has sent the file to the IPND. Each Data Provider will have at least one or a number of different codes making it easily recognisable which Data Provider's source system has sent any particular file to the IPND. There will be no validation by the database to check that the Data Provider code on a public number record matches the File Source Code on the upload file header record. The onus is on the Data Provider to ensure the correct File Source Code is used in the file and that it actually belongs to the Data Provider sending the file. The reason the File Source Code cannot be validated is that a Data Provider may have multiple source systems each with it's own File Source Code and each generating it's own sequence of files.

Explanation of Error
This error is issued when an invalid File Source Code is sent as part of a filename. If we accept XXXXX as a valid File Source Code for a Data Provider, a file would be rejected if it had for example the following invalid filenames.

IPNDUPXX5PX.0000001
IPNDUPABCXX.0000001
IPNDUPX782X.0000001

Corrective Action
This error is corrected by the file being retransmitted with a correct and appropriate File Source code. The File Source code should be valid and should

reflect which source system has provided the data records. If XXXXX is a valid code for a particular Data Provider and XXXXX represents the source system which has sent the data records for that Data Provider, this code would be entered as the filename source code. The above example would be modified to reflect the following:

IPNDUPXXXXX.0000001

3.1.10. 208 - Upload filename source does not match header source

Background

This error code was designed as a form of validation, done by the database in order to ensure that the File Source Code is the same both in the filename and the header record. This assists in confirming that the file has been transmitted intact and helps in confirming the source system which has sent the file.

Explanation of Error

This error is returned when the filename source in the actual filename and filename source in the header sequence do not match. For example:

File Name : IPNDUPXXP3X.0000001
Header : HDRIPNDUPYYYYY000000119980428112233

File Name : IPNDUPXXXXX.0000001
Header : HDRIPNDUPYYYYY000000119980428112233

The filename source in the filename is a different entry to the filename source entered in the header record. This error is returned when the file sent has conflicting information with two different File Source Codes provided for the one file.

Corrective Action

This error is corrected by changing one of the codes. The onus is on the Data Provider to confirm which of the two is the correct File Source Code and to change the field which holds the incorrect code. For this error to be corrected the file needs to be retransmitted with the same File Source Code in both fields and both reflecting the host system sending the file. If we accept that XXXXX is a valid File Source code for the Data Provider concerned, the above examples would be corrected as follows:

File Name : IPNDUPXXXXX.0000001
Header : HDRIPNDUPXXXXX000000119980428112233

3.1.11. 225 - Embedded space(s) found in upload file sequence number

Background

The file sequence number is used to uniquely identify each file provided through a specific File Source code. It is sequential and contiguous and provides a count of all the files sent to the IPND from a particular File Source code.

Explanation of Error
This error is returned when embedded space(s) are found in the seven digit upload sequence number field. Each of the seven digit spaces must be filled with a digit reflecting the number of the file.

IPNDUPXXXXX.000 001	- Embedded space in sequence number
IPNDUPXXXXX.0 0001	- Embedded spaces in sequence number

Corrective Action
This error is corrected by inserting the required digits in the file sequence number. Embedded spaces are not allowed as part of the file sequence number. All the necessary digits must be provided as part of the sequence number. The above examples would be corrected as follows:

IPNDUPXXXXX.0000001	- No embedded spaces in sequence number
IPNDUPXXXXX.0000001	- No embedded spaces in sequence number

3.1.12. 227 - Leading space(s) found in upload file sequence number

Background
The File Sequence Number is used to uniquely identify each file created within the same File Source Code. This error like most of the errors at file level are a mechanism used to confirm that files have been transferred intact without corruption and in the right sequence. To properly identify and validate each file the Sequence Number must be valid and correctly formatted.

Explanation of Error
This error is returned when leading space(s) are found in the seven digit upload Sequence Number field. Leading spaces like embedded spaces are not allowed to be part of the upload File Sequence Number. Each of the seven digit spaces must be filled with a digit representing the number of the file. For example the first file transferred for a particular source code will have six leading zeros followed by the digit 1. Examples of File Sequence Numbers which would generate this error follow:

IPNDUPXXXXX. 00001	- Leading spaces in sequence number
IPNDUPXXXXX. 0001	- Leading spaces in sequence number

Corrective Action
This error is corrected by inserting the required digits in the file sequence number. Leading spaces are not allowed as part of the file sequence number and leading zeros are not provided by the database. All the necessary leading zeros must be provided as part of the sequence number. The above examples would be corrected in the following way:

IPNDUPXXXXX.0000001	- No leading spaces in sequence number
IPNDUPXXXXX.0000001	- No leading spaces in sequence number

3.1.13. 228 - Non digit character(s) found in upload file sequence number

Background
IPND Upload files provided through a specific File Source code are uniquely identified by a Sequence Number. The Sequence Number is a component of the filename and refers to the position of an individual file in relation to the other files provided from this File Source.

Explanation of Error
This error is returned when non-digit characters are found in the seven digit Upload File Sequence Number field. Non-digit characters are not allowed as part of the upload file sequence number. Each of the seven digit spaces must be filled with a digit representing the number of the file. Examples of filenames that would generate this error follow:

IPNDUPXXXXX.00AB001	- Non-digit characters in sequence number
IPNDUPXXXXX.0@00001	- Non-digit characters in sequence number

Corrective Action
This error is corrected by replacing the non-digit characters in the file sequence number with digits. Leading spaces, leading zeros and non-digit characters are not allowed as part of the file sequence number. All the spaces provided must be filled with digits that will define the sequence number. The above examples would be corrected in the following way:

IPNDUPXXXXX.0000001	- Only digits in sequence number
IPNDUPXXXXX.0000001	- Only digits in sequence number

3.1.14. 230 - Trailing space(s) found in upload file sequence number

Background
The File Sequence Number is used to uniquely identify each file created within the same File Source Code. It is important for this reason that the File Sequence Number is valid and correctly formatted.

Explanation of Error
This error is returned when trailing spaces are found in the seven digit upload sequence number field. Trailing spaces are not allowed as part of the Upload File Sequence Number. Each of the seven digit spaces must be filled with a digit representing the number of the file. Examples of filenames that would generate this error follow:

IPNDUPXXXXX.000001	- Trailing spaces in sequence number
IPNDUPXXXXX.00012	- Trailing spaces in sequence number

Corrective Action
This error is corrected by inserting the required digits in the file sequence number. Trailing spaces are not allowed as part of the File Sequence Number and leading

zeros are not provided by the database. All the trailing spaces must be populated by digits to completed the sequence number entry. The above examples would be corrected in the following way:

IPNDUPXXXXX.0000010	- No trailing spaces in sequence number
IPNDUPXXXXX.0001200	- No trailing spaces in sequence number

3.1.15. 233 - Invalid upload file footer create end date format

Background

This fields contains details of the date and time the creation of the data file was completed. There is a corresponding field in the file header which details the date and time the creation of the data file commenced. These two fields mark the beginning and end of the creation of the data file on the Data Provider's source system. The file footer create end date field is 14 digits in length and follows the format YYYYMMDDHHMMSS.

Explanation of Error

This error is issued when the entry in the file footer create end date field is invalid as it does not follow the above outlined format.

TRL000000119990404 25440000016	- the minutes are missing
TRL0000001199615051122440000016	- the year and month are incorrect
TRL00000010404199911384900000016	- the date is incorrectly positioned

Corrective Action

This error is corrected by providing a valid and correctly formatted date in the file footer create end date field.

TRL0000001199904041225440000016	- date and time correctly formatted
TRL0000001199905051122440000016	- date and time correctly formatted
TRL0000001199904041138490000016	- date and time correctly formatted

3.1.16. 234 - Upload file footer create end date is null

Background

This field contains details of the date and time the creation of the data file was completed. There is a corresponding field in the file header which details the date and time the creation of the data file commenced. These two fields mark the beginning and end of the creation of the data file on the Data Provider's source system. The file footer create end date field is 14 digits in length and follows the format YYYYMMDDHHMMSS.

Explanation of Error

This error indicates that the file creation end date field in the footer is empty ie. there are only spaces in the field as shown below:

TRL0000001 0000016	- the date field is null.

Corrective Action
Null or empty spaces as mentioned above are not allowed in the file creation end field. The error is corrected by putting in a valid entry in the upload file footer create end date field as shown below:

TRL00000011999040411384900000016 - date and time correctly formatted

3.1.17. 236 - Upload file footer contains a negative record count

Background
It is essential for the database to be able to confirm exactly where a file begins and exactly where it ends when files are transmitted, for the purpose of validating that file. The header and trailer records sent as part of a file have the task of beginning and completing a file. The trailer record also details the number of records sent in that particular file. The record count refers to the number of transaction records in the file excluding the header and trailer records.

Explanation of Error
This error is returned when the file record count in the file footer contains an invalid record count and therefore is not a true reflection of the total number of data records sent. The example of a trailer below indicates that 16 data records have been sent in a file. If the file actually contains a different number of records this error would be generated.

TRL00000011999040411384900000016

Corrective Action
This error is corrected by providing a positive record count detailing the number of records sent as part of the file excluding the header and trailer records. If there were a total of 80 data records sent as part of a file this would be the record count for the upload file. This should be reflected in the file trailer record as follows.

TRL00000011999040411384900000080

3.1.18. 237 - Unknown footer record type found in upload file

Background
There are three types of records that the database recognises as part of a file. They are headers, trailers (or footers) and actual data records that have no prefix. The record types ‘HDR’ and ‘TRL’ are abbreviations for header and trailer. These record types are used to provide validation on the file. The header and trailer records are used to verify information such as the file name, the record count and the file sequence number. The record type in the upload file trailer record occupies the first three characters of the line and refers to the type of record being sent, in the case of this error this would be the trailer. ‘TRL’ is the standard entry in this field for the trailer record. It is essential for the database to be able to confirm exactly where a file begins and exactly where it ends. The header and trailer serve the purpose of confirming where the transmission begins and ends for a file. This indicator is very important in the validation process of the file as it indicates to the

database that all the data records have been sent and the transmission has ended, as the trailer record is always the last record sent as part of the file.

Explanation of Error
This error is generated when the trailer record type in the trailer is unknown and therefore invalid. The following is an example where this error would be returned.

FTR0000001199904041138490000016

Corrective Action
This error is corrected by the insertion in the field of the only accepted trailer record type which is 'TRL'. The above example would be corrected in the following way:

TRL0000001199904041138490000016

3.1.19. 238 - Invalid character(s) found in upload file footer record count

Background
The footer is an important source of information enabling successful validation of a file. The trailer record is used by the database to confirm that a file has been transmitted in its entirety. The trailer record also has the task of detailing the number of records sent in that particular file. The record count is a field in the footer which refers to the number of data records sent in the file excluding the header and trailer records.

Explanation of Error
The file trailer record count is a 7 digit field reflecting the number of transaction records in the file excluding the header and trailer records. This error is returned when this field is populated with non-digit characters.

TRL0000001199904041138490$0160
TRL0000001199904041138490002O@7
TRL0000001199904041138 49eighteen

Corrective Action
This error is corrected by the provision of a valid and accurate file record count in the file trailer. This figure should be the total sum of all the data records sent as part of the file. The following are corrections of the above examples.

TRL0000001199904041138490000160
TRL0000001199904041138490002087
TRL0000001199904041138490000018

3.1.20. 239 - Upload file trailer record count does not match number of records found in file

Background
It is essential for the database to be able to confirm exactly where a file begins and exactly where it ends when it has been transmitted, for the purpose of validating

that file. The header and trailer records sent as part of a file have the task of beginning and completing a file. The trailer record also details the number of records sent in that particular file. The record count refers to the number of transaction records in the file excluding the header and trailer records. The record count field must contain a valid and accurate entry reflecting the total number of actual data records sent in the file.

Explanation of Error
If the file record count field in the Upload File trailer record does not match the total actual number of data records in the file then the file will be rejected and reported in the error file. If the following trailer record was sent as part of a file that transmitted 3896 data records this error would be returned.

TRL00000011999040411384900001 92

Corrective Action

This error is corrected by the entry in the trailer file record count field of the total number of data records transmitted as part of the file. The above example would be corrected in the following way.

TRL00000011999040411384900038 96

In certain circumstances the conversion of EBDIC characters to ASCII might cause a line to prematurely terminate. This error will need to be recftied by identfying the line

On rare occcassions even though the number of records match the number of records in the trailer, the IPND will reject the file with this error. This may be caused by the conversion of EBDIC characters to ASCII, which inserts a control character between a line and splits them into two. This problem can be resolved by either inster

3.1.21. 240 - Upload file footer record count is null

Background
The details in the trailer record are an important control mechanism used by the database to enable successful validation of a file. The trailer record is used by the database to confirm that a file has been transmitted in its entirety. The file record count is a field in the trailer which refers to the number of data records sent in the file excluding the header and trailer records.

Explanation of Error
This error is returned when the record count field is null (ie. has no value in it). This seven digit field must contain a valid and accurate entry reflecting the total number of data records sent as part of the file. The following is an example of a trailer record where the space for the record count has been left blank.

TRL000000119990404113849

Corrective Action

This error is corrected when the file is resent with a value in the trailer record count specifying the total number of data records sent in the file. If there were a total of 1002 data records in the file sent, the above example would be corrected in the following way:

TRL000000119990404113849000001002

3.1.22. 241 - Number of records exceeds maximum limit

Background
As of 1st December, 2016, the number of data records or rows in a file must be limited to 100,000. This does not included the header and footer rows.
This limit has been introduced to preserve the integrity of the IPND. Very large files can not only potentially cause errors, but also have an effect on downstream users.

Explanation of Error
This error is returned when the records (rows) in the file exceed 100,000
Header and footer records are not counted.

Corrective Action
This error is corrected by the file being resubmitted with 100,000 or fewer data records in the file.
Header and footer are not counted.
Note: as this is a file level error, the file must be submitted again with the same sequence number.

3.1.23. 242 - Invalid character(s) found in upload file footer sequence number

Background
The File Sequence Number is used to uniquely identify each file created within the same File Source Code. The Sequence Number is a component of the filename and refers to the position of an individual file in relation to the other files provided from this File Source. This number is incremented by 1 for each new file created and sent to the database. This error like most of the errors at file level are a mechanism used to confirm that files have been transferred intact and in the right sequence. To properly identify and validate each file the Sequence Number must be valid and correctly formatted.

Explanation of Error
This error is returned when there are invalid characters entered in the file footer sequence number field. Alpha characters and characters outside the valid ASCII

range are not allowed in this field. The following are examples where this error would be generated.

TRL000@001199904041138490000160
TRL00*000*199904041138490002087
TRL=+00001199904041138490000018

Corrective Action
This error is corrected by the file being resubmitted with a correct File Sequence Number

TRL0000001199904041138490000160
TRL0000001199904041138490002087
TRL0000001199904041138490000018

3.1.24. 243 - Upload file footer sequence number is null

Background
The File Sequence Number is used to uniquely identify each file created within the same File Source Code. They represent a unique consecutive number for the file from the source system. This error is a control mechanism used to confirm that files have been transferred in their. To properly identify and validate each file the Sequence Number must be valid and correctly formatted.

Explanation of Error

This error will result and the file will be rejected when the file footer sequence number field has been left blank as indicated in the following example.

TRL 199904041138490000160

Corrective Action
To correct this error the file footer sequence number of the last successfully uploaded file plus 1 is inserted in the file footer sequence number field. If this is the first file to be sent to the IPND production database the file sequence number will be 0000001 as highlighted in the example below.

TRL0000001199904041138490000160

3.1.25. 245 - Invalid upload file header create date format

Background
This fields contains details of the date and time the creation of the data file was completed. There is a corresponding field in the file header which details the date and time the creation of the data file commenced. These two fields mark the beginning and end of the creation of the data file on the Data Provider's source system. The file footer create end date field is 14 digits in length and follows the format YYYYMMDDHHMMSS.

Explanation of Error
This error is issued when the entry in the file footer create end date field is invalid as it does not follow the above outlined format.

HDRIPNDUPXXXXX.00000122804199811223 3

Corrective Action
This error is corrected by providing a valid and correctly formatted date in the file footer create end date field.

HDRIPNDUPXXXXX.00000121998042811223 3

3.1.26. 246 - Upload file header create start date is null

Background
This field contains details of the date and time the creation of the data file commenced. There is a corresponding field in the file trailer which details the date and time the creation of the data file was completed. These two fields mark the beginning and end of the creation of the data file on the Data Provider's source system. The file header create start date field is 14 digits in length and follows the format YYYYMMDDHHMMSS.

Explanation of Error
This error indicates that the file creation start date field in the trailer is empty ie. there are only spaces in the field as shown below:

HDRIPNDUPXXXXX.0000012

Corrective Action
Null or empty spaces as mentioned above are not allowed in the file creation end field. The error is corrected by putting in a valid entry in the upload file trailer create end date field as shown below:

HDRIPNDUPXXXXX.00000121998042811223 3

3.1.27. 247 - Invalid file source found in upload file header

Background
The File Source Code is a 5 character code embedded in the filename which identifies the Data Provider's source system sending the data. Each Data Provider will have at least one but may have a number of different codes. The File Source code is compared to a database of valid codes to determine the validity of the code. There will be no validation by the database in ensuring that the Data Provider code on a public number record is appropriate for the File Source code in the filename. The onus is on the Data Provider to ensure the correct File Source code is used in the file and that it actually belongs to the Data Provider sending the file. The reason the File Source code cannot be validated against the Data Provider code is due to the fact that a Data Provider may have multiple systems each with it's own File Source code and each generating it's own sequence of files.

Explanation of Error
This error is issued when the File Source code that is provided in the upload file header is checked against the database of valid codes and found to be invalid. If for example Data Provider A has two source systems which provide data to the IPND. The File Source code of one is XXXXX and the code of the other is YYYYY. If Data Provider A tried to send a File Source code of XXXXY or ZZZZZ or some other unrecognised code this error would be returned.

IPNDUPXXXXX.0000001	- Valid code - Accepted
IPNDUPYYYYY.0000001	- Valid code - Accepted
IPNDUPXXXXY.0000001	- Invalid code - Rejected
IPNDUPZZZZZ.0000001	- Invalid code - Rejected

Corrective Action
This error is corrected by the file being retransmitted with a correct and appropriate File Source code. The File Source code should be valid and should reflect which source system has provided the data records. If the source system which has supplied the data records has the File Source code XXXXX, YYYYY though it is a valid code recognised by the IPND, is not an appropriate code to use in this filename as YYYYY does not reflect the source system of the file. A file with a File Source code of YYYYY though would not be rejected as it is a valid source code. The onus is on the Data Provider to ensure the File Source code is valid and appropriate, reflecting the source system which has generated the file.

3.1.28. 248 - Invalid file type found in upload file header

Background
IPND Upload Files must have all the relevant fields populated in the upload file header. This file header is a 35 character string detailing important information about the file. These details provide an important way of validating the file and confirming that the file has been sent intact.

Explanation of Error
This error is returned when the file type field has an entry other than 'UP'. In an upload file the file type field will be populated with 'UP'. Any other entry is considered invalid.

Header : HDRIPNDXXYYYYY00000011998042811223 3
Header : HDRIPND YYYYY00000011998042811223 3
Header : HDRIPND@$YYYYY00000011998042811223 3

Corrective Action
For this error to be corrected the file needs to be resent with a correct and valid file type in the upload file header. The above examples would be corrected in the following way.

Header : HDRIPNDUPYYYYY00000011998042811223 3

3.1.29. 249 - Invalid record type found in upload file header

Background
There are three types of records that the database recognises as part of a file. They are headers, footers (or trailers) and actual data records that have no prefix. The record types 'HDR' and 'TRL' are abbreviations for header and trailer. These record types are used to provide validation on the file. The header and trailer records are used to verify information such as the file name the record count and the file sequence number. They also signal the beginning and end of transmission of a data file.

Explanation of Error
This error is returned when the entered record type is not valid ie. not 'HDR'.

Header : TRLIPNDUPYYYYY0000001199804281122 33
Header : @#RIPNDUPYYYYY0000001199804281122 33

Corrective Action
This error is corrected by resubmitting the file with the correct record type in the file header as shown below.

Header : HDRIPNDUPYYYYY0000001199804281122 33

3.1.30. 250 - Invalid character(s) found in upload file header sequence number

Background
IPND Upload files provided through a specific File Source code are uniquely identified by a Sequence Number. The Sequence Number is a component of the filename and is found in the header and trailer records of the file. It refers to the position of an individual file in relation to the other files provided from this File Source.

Explanation of Error
This error is returned when invalid characters are found in the seven digit file header sequence number field. Invalid characters are not allowed in the header file sequence number field. Each of the seven digit spaces must be filled with a digit representing the number of the file. Examples of filenames that would generate this error follow:

HDRIPNDUPYYYYY0#00@0119980428112233 - Invalid characters
HDRIPNDUPYYYYYABC00119980428112233 - Invalid characters

Corrective Action
This error is corrected by inserting the required digits in the file sequence number and removing any unnecessary and invalid characters. The above example would be corrected in the following way:

HDRIPNDUPYYYYY0000001199804281122 33 - Valid characters
HDRIPNDUPYYYYY0000001199804281122 33 - Valid characters

3.1.31. 251 - Upload file header sequence number is null

Background
The File Sequence Number is used to uniquely identify each file created within the same File Source Code. They are sequential and contiguous. They represent a unique consecutive number for the file from the source system. This error is a control mechanism used to confirm that files have been transferred without corruption and in the right sequence. To properly identify and validate each file the Sequence Number must be valid and correctly formatted.

Explanation of Error

This error will result and the file will be rejected when the file header sequence number field has been left blank as indicated in the following example.

HDR 19990404113849000016 0

Corrective Action
To correct this error the file header sequence number of the last successfully uploaded file plus 1 is inserted in the file header sequence number field. If this is the first file to be sent to the IPND production database the file sequence number will be 0000001 as highlighted in the example below.

HDR0000001199904041138490000160

3.1.32. 252 - Upload file header sequence number does not match footer sequence number

Background
The File Sequence Number is used to uniquely identify each file created within the same File Source Code. They are sequential and contiguous. They represent a unique consecutive number for the file from the source system. The File Sequence Number is a compulsory field in the filename, header and trailer records. This error is a control mechanism used to confirm that files have been transferred without corruption and in the right sequence. To properly identify and validate each file the Sequence Number must be the same in all 3 fields.

Explanation of Error

This error will result and the file will be rejected when the file header sequence number field does not match the trailer sequence number as indicated in the following example.

HDRIPNDUPXXXXX0000099199904041138490000160
TRL0000009199904041138490000160

Corrective Action
To correct this error the file header sequence number of the last successfully uploaded file plus 1 is inserted in the file header sequence number field. This entry must match the entry in the trailer sequence number field also. If this is the first file to be sent to the IPND production database the file sequence number will be 0000001 as highlighted below.

HDRIPNDUPXXXXX0000001199904041138490000160
TRL0000001199904041138490000160

3.1.33. 253 - Upload file footer is too large

Background

To enable accurate and complete validation of files transferred the database is designed to accept records of a fixed length. Receiving records a standard set length allows the database to determine the beginning and end of each record. This predetermined record length allows each record to be processed correctly and in its entirety as it is possible to confirm exactly where the record begins and ends. The standard fixed length of each record is 906 characters. This length is constant to every record in the file regardless of whether it is a header, trailer or data record.

Explanation of Error

This error is generated when the upload file trailer record size is too large and has more characters and or spaces than the allowed 906.

TRL0000001199901012222220000100_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____111112

Corrective Action

The data or spaces in some or all of the fields of the trailer record, which are not required need to be discarded or truncated to the acceptable record length for this error to be cleared. The following is an example of a trailer record that is the acceptable length with no superfluous data or spaces.

TRL0000001199901012222220000100_____
_____
_____
_____
_____
_____
_____
_____
_____
_____

3.1.34. 254 - Upload file footer is too small

Background

To enable accurate and complete validation of files transferred the database is designed to accept records of a fixed length. Receiving records of a standard set length allows the database to determine the beginning and end of each record. This predetermined record length allows each record to be processed correctly and in its entirety. The standard fixed length of each record is 906 characters. This length is constant to every record in the file regardless of whether it is a header, trailer or data record.

Explanation of Error

This error is generated when the upload file trailer record size is too small and is missing either required characters or spaces. The following example is missing the last two digits of the record count and 625 spaces.

TRL0000001199901012222220000 1

Corrective Action

The data or spaces in some or all of the fields of the trailer record, which are missing need to be added to bring the trailer to an acceptable length for this error to be cleared. The following is an example of a trailer record that is the acceptable length with all the required alpha-numeric characters and spaces.

TRL0000001199901012222220000100

3.1.35. 255 - Upload file header is too large

Background

To enable accurate and complete validation of files transferred the database is designed to accept records of a fixed length. Receiving records a standard set length allows the database to determine the beginning and end of each record. This predetermined record length allows each record to be processed correctly and in its entirety. The standard fixed length of each record is 906 characters. This length is constant to every record in the file regardless of whether it is a header, trailer or data record.

Explanation of Error

This error is generated when the upload file header record size is too large and has more characters and or spaces than the allowed 906.

HDRIPNDUPXXXXXX000000119990101222222_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____####$%&

Corrective Action

The data or spaces in some or all of the fields of the header record, which are not required need to be discarded or truncated to the acceptable record length for this error to be cleared. The following is an example of a header record that is the acceptable length with no superfluous data or spaces.

HDRIPNDUPXXXXXX000000119990101222222_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____

3.1.36. 256 - Upload file header is too small

Background

To enable accurate and complete validation of files transferred the database is designed to accept records of a specified fixed length. Receiving records a standard set length allows the database to determine the beginning and end of each record. This predetermined record length allows each record to be processed correctly and in its entirety. The standard fixed length of each record is 906 characters. This length is constant to every record in the file regardless of whether it is a header, trailer or data record.

Explanation of Error

This error is generated when the upload file header record size is too small and is missing either required characters or spaces. The following example is missing the last four digits of the file creation start date and 198 spaces

HDRIPNDUPXXXXXX00000011999010122_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____

Corrective Action

The data or spaces in some or all of the fields of the trailer record, which are missing need to be added to bring the header to an acceptable length for this error to be cleared. The following is an example of a header record that is the acceptable length with all the required alpha-numeric characters and spaces

HDRIPNDUPXXXXXX000000119990101222222_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____

3.1.37. 259 - Unprintable character found in header

Background

Upload files transmitted to the IPND are to be in ‘flat ASCII’ format. All data are to be printable characters in the range ASCII 32 to ASCII 126 inclusive. The only exceptions are to be the record delimiter characters which will be a ASCII 10 (newline). This error will be returned when these unprintable characters are found in the file header.

Explanation of Error

This error will be returned when unprintable characters outside of the allowed range (ASCII 32 to ASCII 126) are found in the file header.

HDRIPNDUPXXXXXX000000119990101222222_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____ ¶Þ _____ Õ _____

Corrective Action

HDRIPNDUPXXXXXX000000119990101222222_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____

3.1.38. 260 - Unprintable character found in footer

Background

Upload Files transmitted to IPND are to be in ‘flat ASCII’ format. All data are to be printable characters in the range ASCII 32 to ASCII 126 inclusive. The only exceptions are to be the record delimiter characters which will be an ASCII 10 (newline) at the end of each record.

Explanation of Error

This error will be returned when unprintable characters outside of the allowed range (ASCII 32 to ASCII 126) are found in the file trailer.

TRL00000011999010122222200001 00_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____ Œ _____ Œ _____ Õ

Corrective Action

TRL00000011999010122222200001 00_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____

3.2. Mandatory Hard Errors

File level errors are a method by which the database confirms that a file has been transferred intact and in the right sequence. Errors resulting from the attempted loading of an upload file which has inconsistencies, at a file level, will result in an error code being written to the error file indicating that the upload file was rejected and no records were processed in the IPND. As validation at the file level will be done prior to checking at the record or field level, no records will be processed until all the errors at file level have been corrected and the file has been resent.

Once this has been done and there are no longer errors at the file level the database will begin to process each individual data record. The database will check the data records for completeness and accuracy. Essential data that is missing will cause the record to be returned to the Data Provider with a Mandatory Hard error.

Essential data that has been invalidly entered will also cause the record to be returned to the Data Provider with a Mandatory Hard error. This includes numeric fields containing non-numeric data and non-numeric fields containing numeric data. If for example the public number has been entered in the locality field this will result in a Mandatory Hard error.

These types of Mandatory Hard errors will prevent the uploading of the data record to the IPND. The record will not be processed and will not enter the database until it has been corrected and resent.

Data records that conflict with data already existing in the database may also return a Mandatory Hard Error. These records will consequently also not be processed and uploaded to the database.

An example of a what a data record sent to the IPND can look like follows:

0299999999_____CFFJOHNSMITHPTYLTD_____
__T/AJOHNSMITHSOLUTIONSPTYLTD_____
_____
_____JOHNSMITHPTYLTD_____
_____UNITS_____6A____8CFL___3
ATREASURYBUILDING_____REAR_____
_____15A_____19CSMITH_____ST_____SOUTH

_____SMITHFIELD_____
_____VIC3000_UNITS_____6A____8CFL___3ATREASURYBUILDING__
_____REAR_____15A_____1
9CSMITH_____ST_____SOUTH_____
_____SMITHFIELD_____VIC3000LEBFA
X__SMITH_____JOHN_____
_____0299999999YYYXXXXX_1999010100000
019990101000000T_____0288888888

3.2.1. 005 - Unprintable Character in record

Background
Upload files transmitted to IPND are to be in 'flat ASCII' format. All data entries are to be printable characters in the range ASCII 32 to ASCII 126 inclusive. The only exceptions are to be the record delimiter characters which will be a ASCII 10 (newline).

Explanation of Error
This error is returned when unprintable characters (characters not in the range ASCII 32 to ASCII 126) are found in the data record other than the delimiter character at the end of the record. The database does not recognise any characters outside the above allowed range. These characters, that are outside the allowed range are invalid and unprintable as far as the IPND is concerned

Corrective Action
This error is corrected by removing these characters outside the allowed range from the data record and retransmitting the record without these invalid characters.

3.2.2. 006 - Missing Public Number

Background
The public number is the unique identifier of the service. There may be a number of different data records in the database with the one address and the one company name but there will only be the one record with a status of either connected or disconnected in the database at any given time for a public number.

Explanation of Error
This error is generated when the public number is missing from the data record. In the string displayed below the first 61 characters are shown. The public number is the first field in the data record and should be provided in the first 20 character spaces. As the public number entry must not have any leading spaces as part of the entry, the public number along with the area code must begin in the very first available space in the public number field. In the string displayed below the public number is missing from the record.

_____CFFJOHNSMITHPTYLTD_____

Corrective Action
This error is corrected by the provision of the public number in the first 20 character field in the data record.

0299999999_____CFFJOHNSMITHPTYLTD_____

3.2.3. 007 - Missing service status code

Background
The service status code is a one character field indicating the status of the service record. A data record will have a service status code value that is either 'C' for connected or 'D' for disconnected. Only one current Service record may exist for a Public Number. Therefore there can only be the one record, either reflecting a connected or disconnected status for a public number in the Actual Service Table in the database. The Pending Service Table on the other hand can store two records in the database, a pending connect and a pending disconnect. It is essential to correctly tag a service in order to reflect the true status of the service and enable other validation to be performed.

Explanation of Error
This error is returned when the service status field is left blank. This field is a Mandatory Hard field which requires an entry before the record can be processed. The following example shows a string with the service status code missing.

0299999999_____FFJOHNSMITHPTYLTD_____

Corrective Action
For this error to be corrected the letter 'C' or 'D' is inserted in the service status code field reflecting whether the public number is connected or disconnected.

0299999999_____CFFJOHNSMITHPTYLTD_____

0299999999_____DFFJOHNSMITHPTYLTD_____

3.2.4. 008 - Missing pending flag

Background
The database holds two types of tables, an Actual and a Pending service table. Only one Actual record can exist in the database at any given time, either a connected or disconnected public number record. The database can though accept both a pending connected and a pending disconnected record for a Public Number. For pending services the Public Number and the service status will be the unique identifier. The field is populated with a F for false if the record is an Actual and T for true if the record is Pending. This field must have an entry of F or T which reflects whether the record is actual or pending and which also corresponds with the next field (the cancel pending flag entry). Please refer to error code 102 for details about the entries required in these two fields.

Explanation of Error
This error is returned when the pending flag field is left blank. This field is a Mandatory Hard field which requires an entry before the record can be processed. The following example shows a string with the pending flag code missing.

0299999999_____C_FJOHNSMITHPTYLTD_____

Corrective Action
For this error to be corrected the letter 'T' or 'F' is inserted in the pending flag field reflecting whether the public number record is pending or actual as shown in the examples below.

0299999999_____CTFJOHNSMITHPTYLTD_____
0299999999_____CTTJOHNSMITHPTYLTD_____
0299999999_____CFFJOHNSMITHPTYLTD_____

3.2.5. 009 - Missing cancel pending flag

Background
The database holds two types of tables, an Actual and a Pending service table. Only one Actual record can exist in the database at any given time, either a connected or disconnected public number record. The database can accept one pending Connected and one pending Disconnected record for a Public Number. For pending services the Public Number and the service status will be the unique identifier. This field will be populated with a F for false if a pending record does not exist or if the pending record exists but it is not to be cancelled and T for true if the record is Pending and it is to be cancelled. The entry in this field must correspond with the entry in the field that precedes it. Please refer to error code 102 for details.

Explanation of Error
This error is returned when the cancel pending flag field is left blank. This field is a Mandatory Hard field which requires an entry before the record can be processed. The following example shows a string with the cancel pending flag code missing.

0299999999_____CT_JOHNSMITHPTYLTD_____

Corrective Action
For this error to be corrected the letter 'T' or 'F' is inserted in cancel pending flag field reflecting whether the pending record is to be cancelled or not cancelled as shown in the examples below.

0299999999_____CTFJOHNSMITHPTYLTD_____
0299999999_____CTTJOHNSMITHPTYLTD_____
0299999999_____CFFJOHNSMITHPTYLTD_____

3.2.6. 010 - Missing list code

Background

The list code field contains directory information indicating whether the service is to be listed, unlisted or suppressed address. This information must be provided for every public number record. Entries in this field are verified against a database table which contains all the valid list codes. The codes used in the list code field are ‘LE’ for Listed Entry, ‘UL’ for Unlisted Number (which incorporates silent lines) and ‘SA’ for Suppressed Address.

Explanation of Error
This error is issued when the list code field is empty. This information is regarded as mandatory directory information and must be provided with each data record. In the following string the 2 blank spaces where the list code entry should be, immediately follow the postcode entry.

SMITHFIELD_____VIC3000__BFAX__

Corrective Action
For this error to be corrected the record must be resent with a list code provided as part of the record. The example above would be corrected in the following way:

SMITHFIELD_____VIC3000LEBFAX__

3.2.7. 012 - Missing data provider code

Background
The Data Provider code is an individually assigned 6 character field which reflects the Data Provider’s source system which has supplied the data record. It is important for the IPND during processing to be able to identify and confirm which Data Provider’s Source system has supplied the data record. This enables confirmation that the required sequence number is expected from the correct system.

Explanation of Error
This error is issued when the Data Provider code field is empty. This information is regarded as mandatory information used for validation by the database and must be provided with each data record. In the following string we see 6 blank spaces where the Data Provider code entry should be, immediately preceding the transaction date.

YYY_____19990101000000199901010000 00F_____0288888888

Corrective Action
For this error to be corrected the record must be resent with a Data Provider code provided as part of the record. The example above would be corrected in the following way:

YYYXXXXX_19990101000000199901010000 00F_____0288888888

3.2.8. 013 - Incorrect service status code

Background

The service status code is a one character field indicating the status of the service record. A data record will have a service status code value that is either ‘C’ for connected or ‘D’ for disconnected. Only one current service record may exist for a Public Number. It is essential to correctly tag a service in order to reflect the true status of the service and in order to provide accurate data to all Data Users of the IPND.

Explanation of Error
This error is returned when the service status code field has an entry which is incorrect. The service status code is a Mandatory Hard field which requires a valid entry before the record can be processed. The only valid options in the service status code field are ‘C’ and ‘D’, which indicate whether the public number is a connected or disconnected service.

0299999999_____1TFJOHNSMITHPTYLTD_____

Corrective Action
For this error to be corrected the letter ‘C’ or ‘D’ is to be inserted in the service status code field reflecting whether the service is connected or disconnected. There are no other valid options for this field. In the examples that follow the status field which immediately follows the public number field has a code of ‘C’ inserted in the first string and a code of ‘D’ inserted in the second string.

0299999999_____CTFJOHNSMITHPTYLTD_____

0299999999_____DTFJOHNSMITHPTYLTD_____

3.2.9. 014 - Incorrect pending flag

Background
The database holds two types of tables, an Actual and a Pending service table. Only one Actual record can exist in the database at any given time, either a connected or disconnected public number record. The database can accept one pending Connected and one pending Disconnected record for a Public Number. For pending services the Public Number and the service status will be the unique identifier. The field is populated with a F for false if the record is an Actual and T for true if the record is Pending. This field must have an entry of F or T which reflects whether the record is actual or pending and which also corresponds with the next field (the cancel pending flag entry). Please refer to error code 102 for details.

Explanation of Error
This error is returned when the pending flag field is populated with an incorrect entry. This field is a Mandatory Hard field which requires a valid entry before the record can be processed. The following example shows a string with an incorrect pending flag entry.

0299999999_____CNFJOHNSMITHPTYLTD_____
0299999999_____CXTJOHNSMITHPTYLTD_____

Corrective Action
For this error to be corrected the letter ‘T’ or ‘F’ is inserted in pending flag field reflecting whether the public number is pending or not as shown in the examples below.

0299999999_____CFFJOHNSMITHPTYLTD_____
0299999999_____CTTJOHNSMITHPTYLTD_____
0299999999_____CTFJOHNSMITHPTYLTD_____

3.2.10. 015 - Incorrect cancel pending flag

Background
The database holds two types of tables, an Actual and a Pending service table. Only one Actual record can exist in the database at any given time, either a connected or disconnected public number record. The database can accept one pending Connected and one pending Disconnected record for a Public Number. For pending services the Public Number and the service status will be the unique identifier. This field will be populated with a F for false if a pending record does not exist or if the pending record exists but it is not to be cancelled and T for true if the record is Pending and is to be cancelled. The entry in this field must correspond with the previous error code. Please refer to error code 102 for details.

Explanation of Error
This error is returned when the cancel pending flag field has an incorrect entry. This field is a Mandatory Hard field which requires a valid entry before the record can be processed. The following example shows a string with an incorrect entry in the cancel pending flag field.

0299999999_____CTYJOHNSMITHPTYLTD_____
0299999999_____CFNJOHNSMITHPTYLTD_____
0299999999_____CFXJOHNSMITHPTYLTD_____

Corrective Action
For this error to be corrected the letter ‘T’ or ‘F’ is inserted in the service status code field reflecting whether the pending record is to be cancelled or not as shown in the examples below.

0299999999_____CTFJOHNSMITHPTYLTD_____
0299999999_____CTTJOHNSMITHPTYLTD_____
0299999999_____CFFJOHNSMITHPTYLTD_____

3.2.11. 016 - Incorrect list code

Background
The list code field contains directory information indicating whether the service is to be listed, unlisted or suppressed address. This information must be provided for every public number record. Entries in this field are verified against a database table which contains all the valid list codes. The codes used in the list code field

are ‘LE’ for Listed Entry, ‘UL’ for Unlisted Number and ‘SA’ for Suppressed Address. These are the only acceptable entries for the list code field.

Explanation of Error
This error is issued when the list code field is incorrect. If a data record for a Mandatory Hard code field such as the list code has a value that is not valid, the record will be rejected as a hard error and added to the error file. In the following string we see an incorrect entry where the list code entry should be immediately following the postcode entry.

SMITHFIELD_____VIC3000XYBFAX__

Corrective Action
For this error to be corrected the record must be resent with a list code provided as part of the record. The example above would be corrected in one of the following ways.

SMITHFIELD_____VIC3000LEBFAX__
SMITHFIELD_____VIC3000ULBFAX__
SMITHFIELD_____VIC3000SABFAX__

3.2.12. 017 - Incorrect data provider code

Background
The Data Provider code is an individually assigned 6 character field which reflects the Data Provider’s source system which has supplied the data record. It is important for the IPND during processing to be able to identify and confirm which Data Provider’s Source system has supplied the data record.

Explanation of Error
This error is issued when the Data Provider code field has an invalid entry. An incorrect Data Provider code is one that is not valid and not recognised by the IPND This information is regarded as mandatory information used for database validation and must be provided with each data record. In the following string we see 6 blank spaces where the Data Provider code entry should be, immediately preceding the transaction date.

YYY_____19990101000000199901010000 00F_____0288888888

Corrective Action
For this error to be corrected the record must be resent with a valid Data Provider code provided as part of the record. If we take XXXXX to be a valid Data Provider code, the example above would be corrected in the following way:

YYYXXXXX_19990101000000199901010000 00F_____0288888888

3.2.13. 040 - New connected whilst already expecting a Change Data Provider (CDP) disconnect

Background

Due to processing considerations and the fact that connects are usually sent in a more timely manner than disconnects provision has been made in the database for Data Providers to be able to send connection records for public numbers which have already been connected in the IPND by another Data Provider. Currently a Data Provider can overwrite the data record of another Data Provider with their own data record for the same public number. In this scenario a Data Provider sends a connection order which overwrites the details of the existing data record. The service is marked with a flag indicating that the database expects a cancellation order from the previous Data Provider. This causes no error to be generated. The error is generated when the database expecting a cancellation record receives another connect record. For example Data Provider B has sent a connected record replacing Data Provider A's connected record. Data Provider A has not as yet sent a disconnection, but Data Provider C has in the meantime sent a record to connect.

Explanation of Error

This error is generated when a public number has been tagged with a CDP disconnect flag indicating that a disconnection order is expected but in the meantime another Data Provider has sent a connection order. The first connection has not been reconciled but another has already arrived from another Data Provider.

Public Number	Status of Record	DP
02 99999999	C	A
02 99999999	C	B
02 99999999	C	C

In this example a disconnection order from A is still expected by the database so it can reconcile the overwriting of Data Provider A's data by Data Provider B. The database cannot accept another overwriting until the first one has been reconciled.

Corrective Action

For this error to be corrected, Data Provider C which is the carrier that will receive this error will have to confirm whether it owns the service. If the owner is Data Provider B or Data Provider A, Data Provider C will have to ensure their connection is not resent. The error can then be disregarded by Data Provider C. If the owner is Data Provider C, Data Provider A will have to send a disconnect and Data Provider C will have to resubmit their connect once Data Provider A's disconnection has been processed by the database. It would also be preferable if Data Provider B were to send a disconnect but their missing disconnect will not cause an error to be generated. The example below is a possible resolution of this error.

Public Number	Status of Record	Data Provider
02 99999999	C	A
02 99999999	C	B
02 99999999	D	A
02 99999999	C	C

The result in this example is that Data Provider C's data record has overwritten Data Provider B's record. Data Provider B will now be expected to send a disconnection record before the data can be overwritten again.

In this scenario, Data Provider C should contact the IPND Manager who will contact Data Provider A and request a cancellation order be sent to the IPND so that Data Provider C can upload their record which will overwrite Data Provider B's existing connect record.

3.2.14. 041 - Change Data Provider (CDP) disconnect attempted

Background
The database has been designed to perform important validation to ensure the data received is valid and accurate. This error code is an attempt to prevent the unauthorised disconnection of services.

Explanation of Error
This error is issued when a Data Provider tries to disconnect a service that is owned by a different Data Provider. For example Data Provider A has a connected record but Data provider B is trying to disconnect the service in the IPND which is owned by A.

Public Number	Status	Data Provider
02 99999999	C	A
02 99999999	D	B

Corrective Action
The remedy is for Data Provider B to disregard the error and not attempt to resend the disconnection record. They may wish to investigate why the record has been sent to the IPND. There are various reasons why this disconnect may have been sent. It may be a case of an Actual Disconnect having been sent instead of a Pending Disconnect. It may be a case of an old disconnect which is no longer required, being sent.

3.2.15. 042 - Pending service from a different data provider

Background
The management of Pending services can be quite complex so more than two pending transactions are not allowed by the database. The IPND allows two Pending Service records per public number. There may be one Connected and one Disconnected Pending Service record. This error code denotes the receipt by the IPND of two Pending Connect or two Pending Disconnect transactions. Unlike the 'Actual' Services table where 'CDP' transactions are accepted, so that one connect can replace another connect, these transactions are not allowed in the Pending Service table.

Explanation of Error
This error is issued when a pending connect or pending disconnect order is

received by the database from a Data Provider but one already exists in the database for another Data Provider. If a Pending Service exists from Data Provider A with a Status of Connected then the IPND does not allow that it be replaced by a Pending Service from Data Provider B also with a Status of Connected until Data Provider A cancels its pending entry.

Public Number	Status	Data Provider	Pending
02 99999999	C	A	T
02 99999999	C	B	T

Corrective Action
The solution is for Data Provider B to confirm whether Data Provider A's pending connect is still valid. If it is not Data Provider A will have to send a record withdrawing their Pending Connect record. Data Provider B will then be able to resend their pending connect record which will be successfully uploaded to the database. If Data Provider A's pending connect is still valid then Data Provider B will have to ensure their pending connect record is not resent and the error can be ignored ie. considered actioned.

Public Number	Status	Data Provider	Pending
02 99999999	C	A	T
02 99999999	D	A	T
02 99999999	C	B	T

3.2.16. 100 - Public number includes leading spaces

Background
The public number field is 20 spaces and the entry in this field can be up to 20 digits long. One of the requirements of the IPND is that it will not store leading zeros for numeric fields. As the public number entry must not have any leading spaces as part of the entry, the public number along with the area code must begin in the very first available space in the public number field.

Explanation of Error
This error will be returned if the public number entry has leading spaces.

_____0299999999CTTJOHNSMITHPTYLTD_____

Corrective Action
This error is corrected by removing any leading spaces in the public number field.

0299999999_____CTFJOHNSMITHPTYLTD_____

3.2.17. 101 - Public Number includes embedded spaces

Background
The IPND is programmed to accept entries in numeric fields with no leading zeros or embedded spaces. This aspect of the validation process is an important method

for maintaining data integrity and consistency. The entry in the public number field must have no embedded spaces. The public number field is 20 spaces and the entry in this field can be up to 20 digits long.

Explanation of Error
This error is issued when the public number has been entered with embedded spaces. The inclusion of embedded spaces in the public number field will result in this Mandatory Hard error.

02 9999 9999_____CTFJOHNSMITHPTYLTD_____

Corrective Action
This error is corrected by resubmitting this data record with all the embedded spaces removed from the public number field. The above example would be corrected in the following way.

0299999999_____CTFJOHNSMITHPTYLTD_____

3.2.18. 102 - Cancel pending service flag set to true for non pending service

Background
The database holds two types of tables, an Actual and a Pending service table. Only one Actual record can exist in the database at any given time, either a connected or disconnected public number record. The database can accept two pending records, one pending Connected and one pending Disconnected record for a Public Number. For pending services the Public Number and the service status will be the unique identifier.

Explanation of Error
This error is generated when there is an inconsistency in the data provided in the pending and cancel pending flag fields. This error refers to the last example in the truth table below. In this error code the cancel pending flag field is set to 'T' for true indicating that the pending record for the public number is to be cancelled, but the pending flag field is marked with an 'F' for false indicating there is no pending transaction for the public number in the database. This is an incompatible combination of flags as you cannot request the cancellation of a pending record which you believe does not exist in the database.

Transaction	***Pending Flag***	***Cancel Pending Flag***	***Valid***
Pending Connect or Pending Disconnect	T	T	Yes
	T	F	Yes
	F	F	Yes
	F	T	No

0299999999_____CFTJOHNSMITHPTYLTD_____

This string reflects the combination of pending and cancel pending flag fields that would generate this error.

Corrective Action
This error is corrected by the alteration of one or both of the entries in these fields to one of the first three options in the truth table above as shown below:

0299999999_____CTTJOHNSMITHPTYLTD_____
0299999999_____CTFJOHNSMITHPTYLTD_____
0299999999_____CFFJOHNSMITHPTYLTD_____

3.2.19. 105 - Pending service cancel for a non pending service

Background
The database holds two types of tables, an Actual and a Pending service table. Only one Actual record can exist in the database at any given time, either a connected or disconnected public number record. The database can though accept two pending records, one pending Connected and one pending Disconnected record for a Public Number. For pending services the Public Number and the service status will be the unique identifier. For the effective management of pending transactions, all data records (pending and actual) must correctly identify the pending status of the public number and whether or not this status is to be cancelled.

Explanation of Error
This error is issued when there is an entry of ‘T’ in the pending field and ‘T’ in the cancel pending field but there is not actual pending record in the database. The combination of ‘T’ and ‘T’ is an acceptable combination as per the truth table displayed in error code 102. This error though is generated when no pending record actually exists in the database for the public number.

0299999999_____CTTJOHNSMITHPTYLTD_____

Corrective Action
This error is rectified by modifying the pending and cancel pending flag fields to reflect the true pending status of the record in the database. The error in this example would be corrected in the following way.

0299999999_____CFFJOHNSMITHPTYLTD_____

3.2.20. 110 - Non numeric in public number

Background
The public number is the unique identifier of the service. There may be a number of different public numbers with the one address or the one company name but there will only be the one record with a status of connected in the database at any given time for any public number.

Explanation of Error
This error is generated when the public number has non-numeric characters as part of the record. In the string displayed below the first 61 characters are shown. The

public number is the first field of the data record and should be provided in the first 20 character spaces. In this example the public number field contains non-numeric characters which are not allowed and will result in this Mandatory Hard error being returned.

02*99#9999&__*_____CFFJOHNSMITHPTYLTD_____

Corrective Action
This error is corrected by the exclusion of any non-numeric characters from the public number entry. As the public number entry must not have any leading spaces as part of the entry, the public number along with the area code must begin in the very first available position in the public number field and must contain only numeric characters.

0299999999_____CFFJOHNSMITHPTYLTD_____

3.2.21. 257 - Upload file record is too large

Background
Each IPND file record has a fixed length of 906 characters. This length is constant to every record in the file regardless of whether it is a header, trailer or data record. This consistency enables complete and accurate validation to be done by the database as it clearly identifies where a record begins and ends.

Explanation of Error
This error is generated when the upload record size is too large ie. contains more than 906 characters.

Corrective Action
The data in some or all of the fields has to be discarded or truncated to the acceptable record length for this error to be cleared. The following data string is an example of a data record which is the correct required length.

0299999999_____CFFJOHNSMITHPTYLTD_____
__T/AJOHNSMITHSOLUTIONSPTYLTD_____
_____
_____JOHNSMITHPTYLTD_____
_____UNITS_6____A8____CFL3___
ATREASURYBUILDING_____REAR_____
_____15___A__19___CSMITH_____ST_____SOUTH
_____SMITHFIELD_____
_____VIC3000_UNITS_6____A8____CFL3___ATREASURYBUILDING__
_____REAR_____15___A__19__
_CSMITH_____ST_____SOUTH_____
_____SMITHFIELD_____VIC3000LEBFA
X__SMITH_____JOHN_____
_____0299999999_____YYYXXXXX_1999010100000
019990101000000T0288888888_____

3.2.22. 258 - Upload file record is too small

Background

Each IPND file record has a fixed length of 906 characters. This length is constant to every record in the file regardless of whether it is a header, trailer or data record. This consistency enables complete and accurate validation to be done by the database as it clearly identifies where a record begins and ends.

Explanation of Error

This error is therefore generated when the upload file record size is too small ie. contains less than the required 906 characters.

Corrective Action

This error is corrected by the inclusion of the necessary spaces or alpha-numeric characters in the record, to bring the size to the acceptable fixed record length. The following data string is an example of a data record which is the correct required length.

0299999999_____CFFJOHNSMITHPTYLTD_____
__T/AJOHNSMITHSOLUTIONSPTYLTD_____
_____
_____JOHNSMITHPTYLTD_____
_____UNITS_6____A8____CFL3___
ATREASURYBUILDING_____REAR_____
_____15___A__19___CSMITH_____ST_____SOUTH
_____SMITHFIELD_____
_____VIC3000_UNITS_6____A8____CFL3___ATREASURYBUILDING__
_____REAR_____15___A__19__
_CSMITH_____ST_____SOUTH_____
_____SMITHFIELD_____VIC3000LEBFA
X__SMITH_____JOHN_____
_____0299999999_____YYYXXXXX_1999010100000
019990101000000T0288888888_____

3.2.23. 261 - Listed to Unlisted attempted without a disconnect

Background

This error has been designed to cater for changes to directory listing details from listed to unlisted. A customer may request a Data Provider to change the directory listing of a service number, which already exists in the IPND and in the Directory Publishers database as a listed service. The provision by Data Providers of a modification record changing only the directory details of a public number, though changing the IPND, does not provide a way for Directory Publishers to be advised of this change. The establishing of a Mandatory Hard 261 error means a Data Provider will not be able to modify the list code field for a record without a disconnect order. The generation of a 261 error requires a disconnection record to be sent cancelling the service. A connection order is then to be sent with the new directory details before the database can be updated with the new directory details. The mandatory requirement of a disconnection record ensures the Directory

Publishers are advised of the need to exclude these services from their database. This ensures the Directory Publishers are advised of the need to exclude these services from their database.

Explanation of Error
There are two different cases where the error 261 is generated. Examples of both follow:

Service	Data Provider	Status	List code
0299999999	A	C	LE
0299999999	A	C	UL
0299999999	A	C	LE
0299999999	B	C	UL

In the first example we have a Data Provider attempting to change the list code for one of their public numbers. In the second example we have a Data Provider attempting to overwrite an existing data record of another Data Provider *and* to change the directory listing. In both examples, a disconnection order is required to enable these changes to be communicated to Directory Publishers.

Corrective Action
This Mandatory Hard error is corrected by the submission of disconnect and connect records to the database. The Data Provider with the existing record in the IPND sends a disconnect record to the IPND. The Data Provider who owns the service will then send a connection order with the new list code of UL. The above examples would be corrected in the following way.

Service	Data Provider	Status	List code
0299999999	A	C	LE
0299999999	A	D	LE
0299999999	A	C	UL
0299999999	A	C	LE
0299999999	A	D	LE
0299999999	B	C	UL

3.3. Mandatory Soft Errors

Once the initial validation has been done at the first stage and there are no longer errors at the file level the database is ready to begin processing each individual data record. The database will check each data record for completeness and accuracy. It will first check the record for any Mandatory Hard errors to establish whether there are any major problems with the record.

Essential data that is missing, has been invalidly entered or data that conflicts with a data record that already exists in the database will cause the record to be rejected and returned to the Data Provider with a Mandatory Hard Error.

Mandatory Hard Errors will prevent the uploading of the data record to the IPND. The record will not be processed and will not enter the database until it has been corrected and resent.

Once the record has been corrected and resent and it is established that there are no Mandatory Hard Errors for the record, the database will then proceed to validate the record for Soft Errors.

Soft errors refer to Mandatory Soft fields in a record that must contain a valid value but are either empty, have an invalid entry or one that is incompatible. These records wherever possible will be processed by the IPND but a soft error will be returned.

The fields that have passed validation will be uploaded to the database. The fields in question (that have generated soft errors) will where possible be entered into the IPND. The record in the IPND will be set with a Soft Error flag and a Soft Error will be returned to the Data Provider

3.3.1. 020 - Missing customer name 1

Background

Data Provider's have a number of fields where the customer name details can be entered. These are the customer name 1, customer name 2, long name and finding name fields. As there are a number of customer name fields in various parts of the data record the customer name 1 field has been made a Mandatory Soft field.

Explanation of Error

An omission of an entry in the customer name 1 field will result in this Mandatory Soft error being returned.

0299999999_____CFFJOHN_SMITH_PTY_LTD_____
__T/A_JOHN_SMITH_SOLUTIONS_PTY_LTD_____
_____
_____JOHN_SMITH_PTY_LTD_____

Corrective Action

All cases where this soft error has been returned should be assessed to determine why the customer name is missing from this field. Though the customer name can be entered in any one of the other fields, the customer name 1 field is Mandatory Soft and should be populated for this error to be cleared.

0299999999_____CFFJOHN_SMITH_PTY_LTD_____
__T/A_JOHN_SMITH_SOLUTIONS_PTY_LTD_____
_____
_____JOHN_SMITH_PTY_LTD_____

3.3.2. 026 - Missing usage code

Background

The usage code is a Mandatory Soft field which refers to the type of business the service number is used for. The field is a one character field populated by an alpha character. The options are G for government, B for business, R for Residential, C for Charity and N for Not Available.

Explanation of Error
This error is returned when the usage code has been left out from the data record. This field is Mandatory Soft so it will not cause the record to be rejected. The record details will be uploaded to the database but this soft error will be returned for the missing usage code.

_____SMITHFIELD_____VIC3000LE_FAX

Corrective Action
This error is corrected by the provision of a usage code immediately following the list code.

_____SMITHFIELD_____VIC3000LEBFAX

3.3.3. 027 - Missing carriage service provider code

Background
The carriage service provider code refers to a 3 character alpha-numeric code allocated to each Data Provider. This code should be provided as part of the record. It provides an extra form of validation on the record and confirms along with the Data Provider code which Data Provider has sent the record.

Explanation of Error
This error is returned when the carriage service provider code is missing from the record. The other record details as far as possible will be uploaded to the database but the record will be marked with a soft error flag

0299999999_____XXXXX_19990101000000199901010000000F02888
88888_____

Corrective Action
This error is corrected by the provision of the three character carriage service provider code in the data record immediately preceding the data provider code.

0299999999_____YYYXXXXX_19990101000000199901010000000F0288
888888_____

3.3.4. 028 - Missing transaction date

Background
The transaction date is a 14 digit numeric field referring to the date the record was created on the Data Provider's system. This date should refer to the date and time the current service status was reached. For a record that was processed as part of the initial upload the date will be the activation date. For a record that was processed as part of the daily incremental upload it will be the processing date.

Explanation of Error

This error is returned when the transaction date is missing from the record. The omission of the transaction date will not cause the record to be rejected but the transaction date should be provided as part of the record.

0299999999_____YYYXXXXX_____19990101000000F0288
888888_____

Corrective Action
This error is corrected by the provision of a valid transaction date in the transaction date field immediately following the Data Provider code.

0299999999_____YYYXXXXX_19990101000000199901010000 00F0288
888888_____

3.3.5. 029 - Missing service status date

Background
The service status date may be the same date as the transaction date. The service status date is a 14 digit numeric field referring to the date the record was created on the Data Provider's system. This date should refer to the date and time the current service status was reached. For a record that was processed as part of the initial upload the date will be the activation date. For a record that was processed as part of the daily incremental upload it will be the processing date. If the pending flag is set to F the date and time on which the current service status was reached will be the service status date. If the pending flag is set to T the service status date will be the date and time on which the pending service status is expected to be reached.

Explanation of Error
This error is returned when the service status date is missing from the record.

0299999999_____YYYXXXXX_19990101000000_____F0288
888888_____

Corrective Action
This error is corrected by the provision of a valid entry in the service status date field reflecting the date the record was sourced from the source system. The service status date field immediately follows the transaction date field.

0299999999_____YYYXXXXX_19990101000000199901010000 00F0288
888888_____

3.3.6. 030 - Missing alternate address flag

Background
The alternate address flag field indicates if the end point for the carriage service is at an address other than the service address. This is most likely to be used for PABX numbers where a caller could be in a different building to the address of

the prime PABX number. If the address for the service number is different to the site address the alternate address flag must be set to T for true. If the address for the service number is the site address the alternate address flag must be set to F for false.

Explanation of Error
This error refers to there being no entry in the field marking the record as true or false regarding alternate address details. This field must be entered with either an F for false or a T for true.

0299999999_____YYYXXXXX_19990101000000199901010000000_0288888888_____

Corrective Action
This error is corrected by the provision of a alternate address flag in the alternate address flag field immediately following the service status date.

0299999999_____YYYXXXXX_19990101000000199901010000000F0288888888_____

3.3.7. 031 - Missing finding name

Background
The directory finding name field must have a valid entry if the data record List Code is Listed Entry. The finding name is the name which will list the customer details for directory purposes.

Explanation of Error
This error is generated when the finding name is missing from the data record for all services with a list code of LE.

0299999999_____CFFJOHN_SMITH_PTY_LTD_____
__T/A_JOHN_SMITH_SOLUTIONS_PTY_LTD_____
_____
_____

Corrective Action
This error is corrected by the provision of a name in the finding name field. This field is Mandatory Soft if the list code is marked as Listed Entry or Suppressed Address.

0299999999_____CFFJOHN_SMITH_PTY_LTD_____
__T/A_JOHN_SMITH_SOLUTIONS_PTY_LTD_____
_____
_____JOHN_SMITH_PTY_LTD

3.3.8. 033 - Missing directory address locality

Background

The directory address fields are used by directory publishers for listing purposes. This error is one of a number of Soft Errors which refer to fields on which the entry is dependent on the entry in another field. If the public number is to be listed and the data record list code is Listed Entry or Suppressed Address the directory address locality field will be a Mandatory Soft field and will require a valid entry. If the public number is not to be listed and the list code enrty is UL for Unlisted the directory address details will not be required and this error will not be returned

Explanation of Error
This error is returned when the list code is Listed Entry or Suppressed Address for a data record but the directory address locality has not been provided. The string below is an example where the list code is 'LE' but there is no entry in the directory address locality field.

_____VIC3000LEBFAX

Corrective Action
This error is corrected by the provision of an entry in the directory address locality field if the list code is 'LE' or 'SA'. The above example would be corrected in either of the following ways depending on whether the service is to be listed or not.

_____SMITHFIELD_____VIC3000**LE**BFAX
_____**UL**BFAX

3.3.9. 034 - Missing directory address state

Background
The directory address fields are used by directory publishers for listing purposes. This error is one of a number of Soft Errors which refer to fields on which the entry is dependent on the entry in another field. If the public number is to be listed and the data record list code is Listed Entry or Suppressed Address the directory address state field will be a Mandatory Soft field and will require a valid entry.

If the public number is not to be listed and the list code is UL for Unlisted the directory address details will not be required and this error will not be returned.

Explanation of Error
This error is returned when the list code is Listed Entry or Suppressed Address for a data record but the directory address state has not been provided. The string below is an example where the list code is 'LE' but there is no entry in the directory address state field.

_____SMITHFIELD_____3000LEBFAX

Corrective Action
This error is corrected by the provision of an entry in the directory address state field if the list code is 'LE' or 'SA'. The above example would be corrected in either of the following ways depending on whether the service is to be listed or not.

_____SMITHFIELD_____VIC3000**L**EBFAX
_____**UL**BFAX

3.3.10. 035 - Missing directory address postcode

Background
The directory address fields are used by directory publishers for listing purposes. This error is one of a number of Soft Errors which refer to fields on which the entry is dependent on the entry in another field. If the public number is to be listed and the data record list code is Listed Entry or Suppressed Address the directory address postcode field will be a Mandatory Soft field and will require a valid entry. If the public number is not to be listed and the list code is UL for Unlisted the directory address details will not be required and this error will not be returned.

Explanation of Error
This error is returned when the list code is Listed Entry or Suppressed Address for a data record but the directory address postcode has not been provided. The string below is an example where the list code is 'LE' but there is no entry in the directory address locality field.

_____SMITHFIELD_____VIC____LEBFAX

Corrective Action
This error is corrected by the provision of an entry in the directory address postcode field if the list code is 'LE' or 'SA'. The above example would be corrected in either of the following ways depending on whether the service is to be listed or not.

_____SMITHFIELD_____VIC3000**LE**BFAX_
_____**UL**BFAX

3.3.11. 036 - Incorrect usage code

Background
The usage code refers to the type of business the service number is used for. The field is a one character field populated by an alpha character. The options are 'G' for government, 'B' for business and 'R' for residential, 'C' for Charity and 'N' for Not available.

Explanation of Error
This error is returned when there is an incorrect entry in the usage code field.

_____SMITHFIELD_____VIC3000LEPFAX
_____SMITHFIELD_____VIC3000LE*FAX
_____SMITHFIELD_____VIC3000LE4FAX

Corrective Action

This error can be corrected by the provision of a valid usage code of ‘G’, ‘B’, ‘R’, ‘C’ or ‘N’. The example below is for a business customer.

_____SMITHFIELD_____VIC3000LEBFAX

3.3.12. 037 - Incorrect carriage service provider code

Background

The carriage service provider code is a field which takes 3 alpha-numeric characters indicating the carriage service provider who owns the record for the public number. Each Data Provider will have their own carriage service provider code which will reflect ownership of the public number. An incorrect carriage service provider code is one that is not valid and not recognised by the IPND. There will be no validation by the database to ensure that the carriage service provider code on a public number record matches the file source field on the upload file header record. The onus is on the Data Provider to ensure the correct carriage service provider code is used both in the public number records and in the filename and that they both belong to the provider sending the file.

Explanation of Error

This error is returned when an incorrect service provider code is provided as part of a record. Incorrect here denotes a CSP code which does not exist in the table in the database and is not recognised by the IPND.

0299999999_____*&%XXXXX_19990101000000199901010000000_0288888888_____

Corrective Action

This error is corrected by the provision of a valid CSP code. If we take YYY as a valid CSP code the above example would be corrected in the following way.

0299999999_____YYYXXXXX_19990101000000199901010000000_0288888888_____

3.3.13. 038 - Incorrect alternate address flag

Background

The alternate address flag field indicates that the end point for the carriage service may be at an address other than the service address. This is most likely to be used for PABX numbers where a caller could be in a different building to the address of the prime of the PABX numbers.

Explanation of Error

This error refers to an incorrect entry in the alternate address flag field marking the record as true or false regarding it’s alternate address details. If the field has an entry of (F) for false but there is an address in the alternate address fields this error will result. Conversely if the field has an entry of (T) for true but there is no address in the alternate address field this error will result.

Corrective Action
This error is corrected by the provision way to correct this error is to change the flag to (T) if the alternate address fields are populated or to remove the details and leave the flag set to (F).

3.3.14. 047 - Missing customer contact name 1

Background
The customer contact name 1 field refers to the name of the contact person for a public number which is located at an alternate address. This alternate address will be different from the address supplied in the service address field. If the alternate address flag is set to true (T) for a record the customer contact name 1 field must have a valid entry.

Explanation of Error
This error is returned when the alternate address flag is set to 'T' for true for a record but the customer contact name 1 field is missing an entry.

_____
_____YYYXXXXX_19990101000000199
90101000000T0288888888_____

Corrective Action
This error is corrected by providing a valid entry in the customer contact name 1 field if the alternate address flag is set to true. If the public number does not have an alternate address, the alternate address flag must be set to 'F' for false. This error is Mandatory Soft where the alternate address code is set to 'T' for true. This example can be corrected in 1 of 2 ways as appears below.

__*SMITH*_____*JOHN*_____
_____0299999999_____YYYXXXXX_199901010000001
9990101000000**T**0288888888_____

_____
_____YYYXXXXX_199901010000001
9990101000000**F**0288888888_____

3.3.15. 048 - Missing customer contact number

Background
The customer contact number field refers to the contact number details for the customer contact of the public number if the public number is located at a different address to the site address. If the alternate address flag is set to true (T) for a record the customer contact number field must have a valid entry.

Explanation of Error
This error is returned when the alternate address flag is set to 'T' for true for a record but the customer contact number field is missing an entry.

__SMITH_____JOHN_____
_____YYYXXXXX_1999010100000019
9901010000000*T*0288888888_____

Corrective Action

This error is corrected by providing a valid entry in the customer contact name 1 field if the alternate address flag is set to true. If the public number does not have an alternate address, the alternate address flag must be set to 'F' for false. This error is Mandatory Soft where the alternate address code is set to 'T' for true. the above example would be corrected in one of the following ways.

__SMITH_____JOHN_____
_____*0299999999*_____YYYXXXXX_19990101000000
199901010000000*T*0288888888_____

_____
_____YYYXXXXX_19990101000000199
90101000000*F*0288888888_____

3.3.16 050 – Invalid service address postcode

Background

The service address postcode is an important part of the service address details in a data record.

Explanation of Error

The IPND validates the service address postcode against the Australia Post Code file. This error is returned when the service address locality field contains an invalid postcode.

Corrective Action

This error is corrected by populating the service address postcode field with a valid postcode.

Overseas addresses

Please note for overseas addresses the service address postcode field must be populated with "0000" and the service address state with "OS". The IPND will not validate the service address locality, state and postcode fields for overseas addresses.

3.3.17 051 – Invalid Service address state

Background

The service address state is an important part of the service address details in a data record.

Explanation of Error

The IPND validates the service address state against the Australia Post Code file. This error is returned when the service address state field contains an invalid state.

Corrective Action

This error is corrected by populating the service address state field with a valid state.

3.3.18 052 – Invalid service address locality

Background

The service address locality is an important part of the service address details in a data record.

Explanation of Error

The IPND validates the service address locality against the Australia Post Code file. This error is returned when the service address locality field contains an invalid locality

Corrective Action

This error is corrected by populating the service address locality field with a valid locality.

3.3.19 053 – Invalid combination of service address locality, state and Postcode

Background

The service address locality, state and postcode is an important part of the service address details in a data record.

Explanation of Error

The IPND validates the service address locality, state and postcode against the Australia Post Code file. This error is returned when an invalid combination of the state, postcode and locality is provided to the IPND.

Corrective Action

This error is corrected by populating the service address locality, state and postcode fields with a valid combination of the locality, state and postcode.

3.3.20 080 - Non numeric in service address postcode

Background

The service address postcode field is an important part of the service address details in a data record indicating the locality of the public number is located. This field is a four digit numeric field which is Mandatory Soft for every data record.

Explanation of Error
This error is returned when the service address postcode field contains a non-numeric entry. Only numeric entries are accepted as valid in the service address postcode field.

UNITS_6____A8____CFL3___ATREASURYBUILDING_____
_____REAR_____15___A__19___CSMITH_____
_____ST_____SOUTH_____SMITH
FIELD_____VIC*#Tt

Corrective Action
This error is corrected by the provision of a valid and correct postcode entry in the service address postcode field.

UNITS_6____A8____CFL3___ATREASURYBUILDING_____
_____REAR_____15___A__19___CSMITH_____
_____ST_____SOUTH_____SMITH
FIELD_____VIC*3000*

3.3.21 081 - Non numeric in directory address postcode

Background
The directory address postcode field is an important part of the service address details in a data record indicating the locality of the public number. This field is a four digit numeric field which is Mandatory Soft for data records that have a list code of 'LE' or 'SA'.

Explanation of Error
This error is returned when the directory address postcode field contains a non-numeric entry. Only numeric entries are accepted as valid in the directory address postcode field.

_UNITS_6____A8____CFL3___ATREASURYBUILDING_____
_____REAR_____15___A__19___CSMITH_____
_____ST_____SOUTH_____SMIT
HFIELD_____VIC*WWWW*LEBFAX

Corrective Action
This error is corrected by the provision of a valid and correct postcode entry in the directory address postcode field.

UNITS_6____A8____CFL3___ATREASURYBUILDING_____
_____REAR_____15___A__19___CSMITH_____
_____ST_____SOUTH_____SMIT
HFIELD_____VIC*3000*LEBFAX

Where valid postcodes are not available as in the case of international addresses 0000 can be used to populate this field.

3.3.22 082 - Badly formed transaction date

Background
The transaction date refers to the date and time the transaction occurred on the Data Provider's system. For the initial upload into IPND this date will be the activation date. For the daily incremental uploads the date will be the processing date. The transaction date is a fourteen character field. A badly formed transaction date is a date that has not been correctly formatted. The IPND expects a format of yyyymmddhhmmss for example.

Explanation of Error
If the data in this field does not match this format or is invalid this error is issued. As this error is a Mandatory Soft error the record which is issued with this error will be processed and uploaded to the IPND but this particular field may be left blank as the entry is invalid.

_____YYYXXXXX_19990101800101	- hours are incorrect
_____YYYXXXXX_199901010101	- seconds are missing
_____YYYXXXXX_01010101011999	- incorrect format
_____YYYXXXXX_01011999010101	- incorrect format

Corrective Action
This error is corrected by the provision of a valid and correct transaction date. The above examples would be corrected in the following way.

_____YYYXXXXX_19990101010101

3.3.23 083 - Badly formed service status date

Background
The service status date refers to the date the service was connected or disconnected on the Data Provider's provisioning system. For modification orders the original connection date of the service being modified should be used. The service status date takes the form yyyyddhhmmss. The service date usually has no time so the hhmmss spaces should be set to zero. 24:0000 is not acceptable as a time by the database.

Explanation of Error
This error is returned when an entry which does not match the expected format or is an invalid entry is entered in the service status date field.

01010000001999F0288888888_____	- incorrect format
19990101600000F0288888888_____	- incorrect time
19990101 F0288888888_____	- time not set to zero

Explanation of Error
This error is corrected by the provision of a valid service status date. The above examples would be corrected in the following way.

19990101000000F0288888888_____

3.3.24 084 - Missing service address locality

Background

The service address locality is an important part of the service address details in a data record indicating the locality of the public number. This field is 40 characters in length. The locality or suburb should be provided as part of a record for a public number. The omission of the service address locality will not prevent the record from being processed.

Explanation of Error

This error is returned when service address locality has not been provided.

UNITS_6____A8____CFL3___ATREASURYBUILDING_____
_____REAR_____15___A__19___CSMITH_____
_____ST_____SOUTH_____
_____VIC3000

Corrective Action

This error is corrected by the provision of an entry in the service address locality field. This field is Mandatory Soft for all data records. The above example would be corrected in the following way.

UNITS_6____A8____CFL3___ATREASURYBUILDING_____
_____REAR_____15___A__19___CSMITH_____
_____ST_____SOUTH_____*SMITHF*
*IELD*_____VIC3000

3.3.25 085 - Missing service address state

Background

The service address state is an important part of the service address details in a data record indicating the state where the public number is located. This field is 3 characters in length. The state details should be provided as part of a record for a public number.

Explanation of Error

This error is returned when service address state details have not been provided as part of the data record.

UNITS_6____A8____CFL3___ATREASURYBUILDING_____
_____REAR_____15___A__19___CSMITH_____
_____ST_____SOUTH_____SMITH
FIELD_____3000

Corrective Action

This error is corrected by the provision of an entry in the service address state field. This field is Mandatory Soft for all data records. The above example would be corrected in the following way.

UNITS_6____A8____CFL3___ATREASURYBUILDING_____
_____REAR_____15___A__19___CSMITH_____

_____ST_____SOUTH_____SMITH
FIELD_____*VIC*3000

3.3.26 086 - Missing service address postcode

Background
The service address postcode is an important part of the service address details in a data record indicating the locality of the public number. This field is 4 characters in length. The state details should be provided as part of a record for a public number. For those records that do not for whatever reason have a valid postcode, 0000 can be used is the postcode field to prevent a soft error from being generated.

Explanation of Error
This error is returned when service address postcode details have not been provided as part of the data record.

UNITS_6____A8____CFL3___ATREASURYBUILDING_____
_____REAR_____15___A__19___CSMITH_____
_____ST_____SOUTH_____SMITH
FIELD_____VIC____

Corrective Action
This error is corrected by the provision of an entry in the service address postcode field. This field is Mandatory Soft for all data records. The above example would be corrected in the following way.

UNITS_6____A8____CFL3___ATREASURYBUILDING_____
_____REAR_____15___A__19___CSMITH_____
_____ST_____SOUTH_____SMITH
FIELD_____VIC*3000*

3.3.27 103 - Missing directory building property or street name

Background
With different options available for street addressing, sometimes the building property name is enough to successfully identify an address without the provision of a street name as well. In the case of the directory building property and directory street name fields one of the two must be provided if the list code is Listed Entry or Suppressed Address for a data record. The directory building property is a 40 character field which refers to the building or property name of the site. This field is Mandatory Soft if the list code is Listed Entry or Suppressed Address and the Directory Street Name 1 field is empty. Conversely the directory street name field, which is 25 characters in length is Mandatory Soft if the list code is Listed Entry or Suppressed Address and the Directory Building Property field is empty.

Explanation of Error
This error is generated when both the directory building property field and the directory street name field are empty, and the list code is LE or SA for the record.

UNITS_6____A8____CFL3___A_____
_____
_____SMIT
HFIELD_____VIC3000LEBFAX

Corrective Action
For this error to be corrected one of the two fields must be populated with a valid entry.

UNITS_6____A8____CFL3___ATREASURYBUILDING_____
_____
_____SMIT
HFIELD_____VIC3000LEBFAX

UNITS_6____A8____CFL3___A_____
_____REAR_____15___A__19___CSMITH_____
_____ST_____SOUTH_____SMIT
HFIELD_____VIC3000LEBFAX

3.3.28 104 - Missing service building property or street name

Background
With different options available for street addressing, sometimes the building property name is enough to successfully identify a site address without the provision of a street name as well. In the case of the building property and directory street name fields one of the two must be provided for a data record. The directory building property is a 40 character field which refers to the building or property name of the site. This field is Mandatory Soft if the street name 1 field is empty. Conversely the directory street name field, which is 25 characters in length is Mandatory Soft if the building property field is empty.

Explanation of Error
This error is generated when both the building property field and the street name field are empty.

UNITS_6____A8____CFL3___A_____
_____
_____SMIT
HFIELD_____VIC3000_UNITS

Corrective Action
For this error to be corrected one of the two fields must be populated with a valid entry as shown in the examples below.

UNITS_6____A8____CFL3___ATREASURYBUILDING_____
_____
_____SMIT
HFIELD_____VIC3000_UNITS

UNITS_6____A8____CFL3___A_____
_____REAR_____15___A__19___CSMITH_____
_____ST_____SOUTH_____SMIT
HFIELD_____VIC3000_UNITS

3.4. Warnings

The final level of validation done by the database on records received after File, Mandatory Hard and Mandatory Soft errors are Warnings.

Warning error codes highlight that the data associated with a field entry in a data record may be suspect. The record will still be processed and applied to the IPND but a warning will be returned in the error file.

Warnings are usually generated where there may be a potential problem with the data, but it is not sufficiently severe to generate a soft error. Records associated with warnings may not require attention. The onus is on the Data Provider to determine which if any Warning messages may require attention.

3.4.1. 043 - Warning older transaction date

Background
This error refers to a pending connect order that has been sent to the IPND with an old and invalid transaction date. Pending connect records require that a date is provided indicating when in the future the record is likely to become an actual connect.

Explanation of Error
This warning error is issued when the pending connect data record has a date in the transaction date field which has already passed. This error is a warning indicating that the date on the pending connect record is invalid.

0299999999_____CTFJOHNSMITHPTYLTD_____
__T/AJOHNSMITHSOLUTIONSPTYLTD.....

.....0299999999_____YYYXXXXX_19990101000001999010100
0000T0288888888_____

Corrective Action
The Data Provider may wish to update the record with a valid transaction date. This is at the Data Provider's discretion.

0299999999_____CTFJOHNSMITHPTYLTD_____
__T/AJOHNSMITHSOLUTIONSPTYLTD.....

.....0299999999_____YYYXXXXX_19990101000001999010100
0000T0288888888_____

3.4.2. 107 - Prior public number includes leading spaces

Background
When a customer has requested their carrier or carriage service provider to change their service number the carrier or CSP will send a disconnection for the old number and a connection for the new number. The customer's prior number should be specified in the connection transaction to assist in matching the customer's directory information. This is additional information that may be of use to the various Data Users. This error however is only a warning and will not prevent the record from being processed by the database. The prior public number field like the public number field is 20 characters long. The entry of the prior public number must not have any leading spaces as part of the entry.

Explanation of Error
This error is returned if the prior public number does not begin in the very first available space in the prior public number field.

_____0299999999_____YYYXXXX
X_199901010000001999010100000T_____0288888888

_____0299999999_____YYYXXXX
X_199901010000001999010100000T_____0288888888_____

Corrective Action
This error is corrected by removing any leading spaces in the public number field. The above examples would be corrected in the following way.

_____0299999999_____YYYXXXX
X_199901010000001999010100000T0288888888_____

3.4.3. 108 - Prior public number includes embedded spaces

Background
When a customer has arranged through their carriage service provider for their service number to change the Data Provider will send to the IPND a disconnection record for the old public number and a connection record for the new public number. The customer's prior number should be specified in the connection transaction to assist in matching the customer's directory information. This is additional information that may be of use to the various Data Users. This error though is only a warning and will not prevent the record from being processed by the database. The prior public number field like the public number field is 20 characters long. The entry of the prior public number must not have any embedded spaces as part of the entry.

Explanation of Error
This error is returned if the prior public number entry contains embedded spaces.

_____0299999999_____YYYXXXX
X_199901010000001999010100000T02__8888__8888_____

_____0299999999_____YYYXXXX
X_199901010000001999010100000T0_2_8_8_8888_8_8____

Corrective Action
This error is corrected by removing any embedded spaces in the public number field. The above examples would be corrected in the following way.

_____0299999999_____YYYXXXX
X_19990101000000199901010000000T0288888888_____

3.4.4. 109 - Non numeric in prior public number

Background
When a customer has arranged through their carriage service provider for their service number to change the Data Provider will send to the IPND a disconnection record for the old public number and a connection record for the new public number. The customer's prior number should be specified in the connection transaction to assist in matching the customer's directory information. This is additional information that may be of use to the various Data Users. This error is a warning and will not prevent the record from being processed by the database. The prior public number field like the public number field is 20 characters long. The entry of the prior public number must be comprised only of numeric characters.

Explanation of Error
This error is returned if the prior public number entry contains embedded spaces.

_____0299999999_____YYYXXXX
X_19990101000000199901010000000T02###8888%_____

_____0299999999_____YYYXXXX
X_19990101000000199901010000000T02TTTT8888_____

Corrective Action
This error is corrected by removing any non-numeric characters in the public number field. The above examples would be corrected in the following way.

_____0299999999_____YYYXXXX
X_19990101000000199901010000000T0288888888_____

3.5. Obsolete

Error codes marked with an asterisk, denotes error codes which are obsolete. They are essentially error codes that have been replaced by other codes or are no longer required.

The option does exist to activate them in the future to reflect the same or another error type.

A brief description will be given of each error number.

3.5.1. 002 - Mandatory Hard Error

This error has been replaced by individual Mandatory Hard Errors as outlined in preceding pages.

3.5.2. 003 - Mandatory Soft Error

This error has been replaced by individual Mandatory Soft Errors as outlined in preceding pages.

3.5.3. 004 - Warning

This error has been replaced by individual Warning messages as outlined in preceding pages.

3.5.4. 022 - Missing service street name 1

This error refers to the service street name details missing from the record. This error has been replaced with error code 104. This has been done to cater for the those cases where a building property name may be sufficient to identify a site address without a street name having to be provided.

3.5.5. 023 - Missing service street locality

This error refers to the service street locality details missing from the record. This error has been replaced with error code 084.

3.5.6. 024 - Missing service street state

This error refers to the service street locality details missing from the record. This error has been replaced with error code 085.

3.5.7. 025 - Missing service street postcode

This error refers to the service street postcode details missing from the record. This error has been replaced with error code 086.

3.5.8. 032 - Missing directory street name 1

This error refers to street name details missing from the directory section for Listed Entry and Suppressed Address records. This error has been replaced by error code 103.

3.5.9. 044 - Missing building property or street name

This error refers to the building property and street name details missing from the service address section. This error has been replaced by error code 104.

3.5.10. 087 - Invalid State Transitions

State transition refers to a change of status for a public number. A data record will have a Service Status Code value that is either 'C' for connected, 'D' for disconnected, 'PC' for pending connect and 'PD' for pending disconnect. The service status along with the public number is the unique identifier for the public number. There is no field in the IPND representing the last type of transaction processed by the database. This has to be inferred from the existing records. If a Public Number Service Status Code changes from connected to disconnected for example then the disconnect transaction is implied by the transition between states. The pending attribute of a record will be stored in the Pending Flag field. There can be only one of each pending transaction type for any given service, a pending connect and a pending disconnect. State transitions can be quite complex and varied. It would be quite laborious to document every possible permutation and combination of transition changes. It would be simpler to say that all state transitions are valid unless they error with one of the specific error codes already individually covered in the preceding pages. Error code 087 will not be returned.

3.5.11. 150- Badly formed last service transaction date

This error has been made obsolete by the creation of error code 082. If error code 082, which is a Mandatory Soft error is returned for a record, the transaction date field will be null (space filled) as the entry is not valid. Error code 150 can therefore not be returned on the transaction date of a subsequent record as this field will be blank due to the previous error code return.

3.5.12. 209 - Invalid upload filename transmission count format

This error has been replaced by error 236 which is returned on a negative record count in a file trailer.

3.5.13. 226 - Invalid character(s) found in upload file sequence number

This error refers to invalid character(s) being part of the file sequence number. This error is now inactive and has been replaced by error code 228.

3.5.14. 232 - Invalid character(s) found in upload file transmission number

This error refers to invalid character(s) being part of the file transmission number. This error is now inactive and has been replaced by error codes 201, 202, 203 and 204.

3.5.15. 241 - Upload file size is not an exact multiple of the record size

In the initial stages of the design of the database it was found that the standard size for each record would be 99.5 bits. This figure multiplied by the number of records was considered to be the size of the upload file. This has been found to be too restrictive. Provision has since been made for the possibility of an upload file to contain individual data records that are longer or shorter than the standard. This error code has therefore been made obsolete and will not be returned in the error files.

3.5.16. 244 - Upload file footer create end date is null

This error is a duplicate of the file level error 234.

3.6. Process for IPND File Processing

This section will deal with the file processing capabilities of the IPND. It will outline the process usually followed when a file is sent to the IPND by a Data Provider.

3.6.1. File Upload

Data Providers upload a file to the IPND. This file is held within a directory until it is to be processed by the IPND. This file is to conform to the file formats as described in the Technical Requirements for Data Users and Data Providers document.

The file consists of three parts, a header, numerous transactions (data records) and a trailer. If the header does not conform to the file formats description the whole file will be rejected. If individual records are found to contain errors the record is either rejected as a hard error or accepted as a soft error. If the number of records totalled in the trailer and headers records does not equal the number of transaction records contained in the file, the file will be rejected.

Errors are written into the Data Error file as they are detected. This file is posted to a directory for the Data Provider to download.

3.6.2. Error Processing - General Procedure

Data Providers will not be directly notified of any data errors identified during processing. Data Providers are required to connect to the IPND and look for an error file to determine the results of processing the corresponding upload file.

Successful processing of an upload file will generate an error file with a header and trailer but no error records.

Errors resulting from upload file inconsistencies, at a file level, will result in an error code being written to the error file indicating that the upload file was rejected and no changes were applied to the IPND.

Hard errors resulting from upload file inconsistencies at a record or field level will result in the record in question being rejected and written to the error file with an appropriate error code.

Soft errors resulting from upload file inconsistencies at a record or field level, will result in the record in question being applied to the database, tagged as having a soft error, and written to the error file with an appropriate error code.

3.6.3. Validation of upload file

After an upload file has been received and checked for transmission errors it will be moved to a reception location. A copy of the file will be archived to ensure that an unprocessed copy of the file is retained.

The file is at this stage checked for any errors at file level. If there are any errors at file level the file will be rejected.

After the file has been validated it will be moved to a working directory from where it will be processed and loaded into the database. This will ensure that a file is not mistakenly processed more than once.

3.6.4. Data Record Processing

The data record goes through various stages of validation when it is being processed to ensure data integrity and consistency at both the record and field levels.

Hard errors are defined as those that prevent the upload of the record into the IPND. Fields defined as Mandatory Hard must have valid data. Any field that is defined as Mandatory Hard but contains no data will be rejected as a hard error and added to the error file.

If a Mandatory Hard numeric field contains non-numeric data the record will be rejected as a hard error and added to the error file.

If a data record for a Mandatory Hard code field has a value that is not valid then the record will be rejected as a hard error and added to the error file. For example, if the List Code field has a value other than ‘LE’, ‘UL’or ‘SA’ then the record will be rejected as a hard error and added to the error file.

When a soft error occurs the record is loaded into the database with as much data as possible being used. A record for the service with the Soft Error will also be added to the error file. The Soft Error flag of the service record in the IPND will be set to true (T).

A Soft Error will result when a field that is identified as Mandatory Soft is empty (space filled).

If a Mandatory Soft field contains data that is incompatible with the field type a Soft Error will result . The data for that field will not be loaded into the database. For example an incorrectly formatted date will not be loaded but the other details of the record will be uploaded.

If a data record for a Mandatory Soft code field has a value that is not valid then a Soft Error will result. The data for that field will be loaded into the database if consistent with the database structure. If the field cannot be loaded it will be left empty.

Warnings are generated for issues with the data where there may be a potential problem but not sufficiently severe to generate a Soft Error requiring attention.

Records associated with warnings may not require attention.

The Transaction Date refers to the date and time the transaction occurred on the Data Provider’s system.

If the Transaction Date for a data record is earlier than the Transaction Date for an existing record then a Warning is generated and added to the error file. The rationale behind generating a Warning is related to the importance of ensuring the data arrives in the correct sequence. However given the difficulties associated with

data being sourced from multiple systems only a warning is generated for this error.

4. MEASURES

Measures will relate to the number of File, Mandatory Hard and Mandatory Soft errors generated each month and the timeliness with which they are corrected.

It is an aim of this document to enable the timely, efficient and effective correction of errors by all Data Providers

5. REFERENCES

Document Number	Title
	IPND Data Users and Data Providers Technical Requirements for IPND
ACIF G619:2005	Industry Guideline IPND Data

6. DEFINITIONS

The following words, acronyms and abbreviations are referred to in this document.

Term	Definition
IPND	Integrated Public Number Database
CDP	Change Data Provider. Covers scenarios where the Public Number has changed from one Data Provider to another
Connected	The service is connected and operational
Disconnected	The Service has been disconnected and for all intents and purposes is not operational. The soft dial tone facility may be available on a disconnected Service
Pending	An expected change to a Public Number record
Upload	Transfer of a file from an external entity to IPND
Download	Transfer of a file from IPND to an external entity
LNP	Local Number Portability - transferral of service numbers between carriers

7. ATTACHMENTS

Document Number	Title
7.1	Error List

7.1. Error List

Message #	Text	Error Classification
001	File out of sequence.	F
002	Mandatory hard error.	*[1]
003	Mandatory soft error.	*
004	Warning.	*
005	Unprintable character in record	MH
006	Missing public number.	MH
007	Missing service status code.	MH
008	Missing pending flag.	MH
009	Missing cancel pending flag.	MH
010	Missing list code.	MH
012	Missing data provider code.	MH
013	Incorrect service status code.	MH
014	Incorrect pending flag.	MH
015	Incorrect cancel pending flag.	MH
016	Incorrect list code.	MH
017	Incorrect data provider code.	MH
018	Incorrect source ID code.	F
020	Missing customer name1.	MS
022	Missing service street name1.	*
023	Missing service street locality.	*
024	Missing service street state.	*
025	Missing service street postcode.	*
026	Missing usage code.	MS
027	Missing carriage service provider code.	MS
028	Missing transaction date.	MS
029	Missing service status date.	MS
030	Missing alternate address flag.	MS
031	Missing finding name.	MS
032	Missing directory street name1.	*
033	Missing directory address locality.	MS
034	Missing directory address state.	MS

[1] Error codes marked with an asterisk, denotes error codes which are obsolete.

035	Missing directory address postcode.	MS
036	Incorrect usage code.	MS
037	Incorrect carriage service provider code.	MS
038	Incorrect alternate address flag.	MS
040	New connected whilst already expecting a CDP disconnect.	*
041	CDP disconnect attempted.	MH
042	Pending service from a different data provider.	MH
043	Warning older transaction date.	W
044	Missing building property or street name.	*
047	Missing customer contact name1.	MS
048	Missing customer contact number.	MS
050	Invalid service address postcode	MS
051	Invalid service address state	MS
052	Invalid service address locality	MS
053	Invalid combination of service address locality, postcode & state	MS
080	Non numeric in service address postcode.	MS
081	Non numeric in directory address postcode.	MS
082	Badly formed transaction date.	MS
083	Badly formed service status date.	MS
084	Missing service address locality.	MS
085	Missing service address state.	MS
086	Missing service address postcode.	MS
087	Invalid state transition.	*
100	Public number includes leading spaces.	MH
101	Public number includes embedded spaces.	MH
102	Cancel pending service flag set to true for non pending service.	MH
103	Missing directory building property or street name.	MS
104	Missing service building property or street name.	MS
105	Pending service cancel for a non pending service.	MH
107	Prior public number includes leading spaces.	W
108	Prior public number includes embedded spaces.	W
109	Non numeric in prior public number.	W
110	Non numeric in public number.	MH
150	Badly formed lastservice transaction date.	*
201	Incorrect upload filename length.	F
202	Incorrect upload filename prefix.	F
203	Invalid upload filename separator.	F
204	Incorrect upload filename sequence format.	F
205	Upload filename sequence does not match footer sequence.	F
206	Upload filename sequence does not match header sequence.	F
207	Invalid Upload filename source.	F
208	Upload filename source does not match header source.	F

209	Invalid upload filename transmission count format.	*[2]
225	Embedded space(s) found in upload file sequence number.	F
226	Invalid character(s) found in upload file sequence number.	*
227	Leading space(s) found in upload file sequence number.	F
228	Non digit character(s) found in upload file sequence number.	F
230	Trailing space(s) found in upload file sequence number.	F
232	Invalid character(s) found in upload file transmission number.	*
233	Invalid upload file footer create end date format.	F
234	Upload file footer create end date is null.	F
236	Upload file footer contains a negative record count.	F
237	Unknown footer record type found in upload file.	F
238	Invalid character(s) found in upload file footer record count.	F
239	Upload file footer record count does not match number of records found in file.	F
240	Upload file footer record count is null.	F
241	Upload file size is not an exact multiple of the record size.	*
242	Invalid character(s) found in upload file footer sequence number.	F
243	Upload file footer sequence number is null.	F
244	Upload file footer create end date is null.	*
245	Invalid upload file header create date format.	F
246	Upload file header create start date is null.	F
247	Invalid file source found in upload file header.	F
248	Invalid file type found in upload file header.	F
249	Invalid record type found in upload file header.	F
250	Invalid character(s) found in upload file header sequence number.	F
251	Upload file header sequence number is null.	F
252	Upload file header sequence number does not match footer sequence number.	F
253	Upload file footer is too large.	F
254	Upload file footer is too small.	F
255	Upload file header is too large.	F
256	Upload file header is too small.	F
257	Upload file record is too large.	MH
258	Upload file record is too small.	MH
259	Unprintable character found in header.	F
260	Unprintable character found in footer.	F
261	Listed to Unlisted attempted without a disconect.	*[3]

Error codes marked with an asterisk, denotes error codes which are obsolete.
Error codes marked with an asterisk, denotes error codes which are obsolete.

DOCUMENT CONTROL SHEET

Contact for Enquiries and Proposed Changes

If you have any questions regarding this document contact:

Name: Tony Parker

Designation: IPND Manager

Email: ipnd.manager@team.telstra.com

If you have a suggestion for improving this document, complete and forward a copy of *Suggestions for Improvements to Documentation* (form 000 001-F01).

Record of Issues

Issue No	Issue Date	Nature of Amendment
1	31 January 2000	
1.1	17 August 2005	Updated document with Locality, State and Postcode errors
1.2	19th August 2008	Updated IPND Manager details
1.3	15th August 2011	Updated IPND Manager phone number
1.4	July 2019	Updated IPND Manager email

This publication has been prepared and written by Telstra, and is copyright. Other than for the purposes of and subject to the conditions prescribed under the Copyright Act, no part of it may in any form or by any means (electronic, mechanical, microcopying, photocopying, recording or otherwise) be reproduced, stored in a retrieval system or transmitted without prior written permission from the document controller. Product or company names are trademarks or registered trademarks of their respective holders.

Note for non-Telstra readers: The contents of this publication are subject to change without notice. All efforts have been made to ensure the accuracy of this publication. Notwithstanding, Telstra does not assume responsibility for any errors nor for any consequences arising from any errors in this publication.