
i18nspector

Release 0.26

Sep 26, 2020

1	Synopsis	1
2	Description	3
3	Options	5
4	Output format	7
5	Tags	9
5.1	ancient-date	9
5.2	arithmetic-error-in-plural-forms	9
5.3	arithmetic-error-in-unused-plural-forms	9
5.4	boilerplate-in-content-type	10
5.5	boilerplate-in-date	10
5.6	boilerplate-in-initial-comments	10
5.7	boilerplate-in-language-team	10
5.8	boilerplate-in-last-translator	11
5.9	boilerplate-in-project-id-version	11
5.10	boilerplate-in-report-msgid-bugs-to	11
5.11	broken-encoding	11
5.12	c-format-string-argument-type-mismatch	12
5.13	c-format-string-error	12
5.14	c-format-string-excess-arguments	12
5.15	c-format-string-missing-arguments	12
5.16	c-format-string-non-portable-conversion	13
5.17	c-format-string-redundant-flag	13
5.18	codomain-error-in-plural-forms	14
5.19	codomain-error-in-unused-plural-forms	14
5.20	conflict-marker-in-header-entry	14
5.21	conflict-marker-in-translation	14
5.22	conflicting-message-flags	15
5.23	date-from-future	15
5.24	distant-header-entry	15
5.25	duplicate-flag-for-header-entry	15
5.26	duplicate-header-entry	15
5.27	duplicate-header-field	16
5.28	duplicate-header-field-content-transfer-encoding	16

5.29	duplicate-header-field-content-type	16
5.30	duplicate-header-field-date	16
5.31	duplicate-header-field-language	16
5.32	duplicate-header-field-language-team	16
5.33	duplicate-header-field-last-translator	17
5.34	duplicate-header-field-mime-version	17
5.35	duplicate-header-field-plural-forms	17
5.36	duplicate-header-field-project-id-version	17
5.37	duplicate-header-field-report-msgid-bugs-to	17
5.38	duplicate-header-field-x-poedit	17
5.39	duplicate-message-definition	18
5.40	duplicate-message-flag	18
5.41	empty-file	18
5.42	empty-msgid-message-with-plural-forms	18
5.43	empty-msgid-message-with-source-code-references	18
5.44	encoding-in-language-header-field	18
5.45	fuzzy-header-entry	19
5.46	inconsistent-leading-newlines	19
5.47	inconsistent-number-of-plural-forms	19
5.48	inconsistent-trailing-newlines	19
5.49	incorrect-number-of-plural-forms	19
5.50	invalid-content-transfer-encoding	20
5.51	invalid-content-type	20
5.52	invalid-date	20
5.53	invalid-language	20
5.54	invalid-language-team	21
5.55	invalid-last-translator	21
5.56	invalid-mime-version	21
5.57	invalid-mo-file	21
5.58	invalid-range-flag	22
5.59	invalid-report-msgid-bugs-to	22
5.60	language-disparity	22
5.61	language-team-equal-to-last-translator	22
5.62	language-variant-does-not-affect-translation	22
5.63	leading-junk-in-plural-forms	23
5.64	malformed-xml	23
5.65	no-content-transfer-encoding-header-field	23
5.66	no-content-type-header-field	23
5.67	no-date-header-field	24
5.68	no-language-header-field	24
5.69	no-language-team-header-field	24
5.70	no-last-translator-header-field	24
5.71	no-mime-version-header-field	24
5.72	no-package-name-in-project-id-version	25
5.73	no-plural-forms-header-field	25
5.74	no-project-id-version-header-field	25
5.75	no-report-msgid-bugs-to-header-field	25
5.76	no-required-plural-forms-header-field	26
5.77	no-version-in-project-id-version	26
5.78	non-ascii-compatible-encoding	26
5.79	non-portable-encoding	26
5.80	os-error	26
5.81	partially-translated-message	27
5.82	perl-brace-format-string-error	27

5.83	perl-brace-format-string-missing-argument	27
5.84	perl-brace-format-string-unknown-argument	28
5.85	python-brace-format-string-argument-type-mismatch	28
5.86	python-brace-format-string-error	28
5.87	python-brace-format-string-missing-argument	28
5.88	python-brace-format-string-unknown-argument	29
5.89	python-format-string-argument-number-mismatch	29
5.90	python-format-string-argument-type-mismatch	30
5.91	python-format-string-error	30
5.92	python-format-string-missing-argument	30
5.93	python-format-string-multiple-unnamed-arguments	31
5.94	python-format-string-obsolete-conversion	31
5.95	python-format-string-redundant-flag	31
5.96	python-format-string-redundant-length	32
5.97	python-format-string-redundant-precision	32
5.98	python-format-string-unknown-argument	32
5.99	python-format-string-unnamed-plural-argument	32
5.100	qt-plural-format-mistaken-for-c-format	33
5.101	range-flag-without-plural-string	33
5.102	redundant-message-flag	33
5.103	stray-header-line	34
5.104	stray-previous-msgid	34
5.105	syntax-error-in-plural-forms	34
5.106	syntax-error-in-po-file	34
5.107	syntax-error-in-unused-plural-forms	35
5.108	trailing-junk-in-plural-forms	35
5.109	translation-in-template	35
5.110	unable-to-determine-language	35
5.111	unexpected-flag-for-header-entry	35
5.112	unknown-encoding	36
5.113	unknown-file-type	36
5.114	unknown-header-field	36
5.115	unknown-message-flag	36
5.116	unknown-poedit-language	36
5.117	unrepresentable-characters	37
5.118	unusual-character-in-header-entry	37
5.119	unusual-character-in-translation	37
5.120	unusual-plural-forms	37
5.121	unusual-unused-plural-forms	37

6	See also	39
----------	-----------------	-----------

CHAPTER 1

Synopsis

i18nspector [*options*] *file* [*file* ...]

CHAPTER 2

Description

i18nspector is a tool for checking translation templates (POT), message catalogues (PO) and compiled message catalogues (MO) files for common problems. These files are used by the GNU gettext translation functions and tools in many different development environments.

CHAPTER 3

Options

- l lang, --language lang** Assume this language. *lang* should be a 2- or 3-letter ISO 639 language code, possibly followed by underscore and a 2-letter ISO 3166 territory code.
- unpack-deb** Allow unpacking Debian (binary or source) packages.
- j n, --jobs n** Use *n* processes in parallel. *n* can be a positive integer, or `auto` to determine the number automatically. The default is to use only a single process.
- h, --help** Show help message and exit.
- version** Show version information and exit.

CHAPTER 4

Output format

The following format is used for all the reported problems:

code: *file*: *tag* [*extra*]

where:

- *code* is a letter indicating type of the message: **E** (error), **W** (warning), **I** (informative message), or **P** (pedantic message);
- *tag* is a name of the problem that was discovered;
- *extra* can contain additional information about the problem.

5.1 ancient-date

The date refers to the time before the first GNU gettext release. As such, it's extremely unlikely to be correct.

References:

<https://git.savannah.gnu.org/cgit/gettext.git/tree/ChangeLog.0#n1767>

Severity, certainty:

normal, certain

5.2 arithmetic-error-in-plural-forms

Computing a plural form value triggers division by zero or integer overflow. This normally indicates an error in the plural form expression.

References:

https://www.gnu.org/software/gettext/manual/html_node/Plural-forms.html

Severity, certainty:

serious, possible

5.3 arithmetic-error-in-unused-plural-forms

Computing a plural form value triggers division by zero or integer overflow. (But there are no translated messages which use plural forms.) This normally indicates an error in the plural form expression.

References:

https://www.gnu.org/software/gettext/manual/html_node/Plural-forms.html

Severity, certainty:

normal, possible

5.4 boilerplate-in-content-type

The Content-Type header field contains xgettext boilerplate. It should be in the form `text/plain; charset=encoding`.

References:

https://www.gnu.org/software/gettext/manual/html_node/Header-Entry.html#index-encoding-of-PO-files

Severity, certainty:

important, certain

5.5 boilerplate-in-date

The date header field contains xgettext boilerplate. The date format should be `YYYY-MM-DD hh:mm+ZZzz`, e.g. `2011-11-05 10:14+0100`.

Severity, certainty:

normal, certain

5.6 boilerplate-in-initial-comments

The initial comments contain xgettext or msginit boilerplate.

References:

https://www.gnu.org/software/gettext/manual/html_node/Header-Entry.html

Severity, certainty:

minor, possible

5.7 boilerplate-in-language-team

The Language-Team header field contains xgettext boilerplate. It should contain English name of the language, and the email address or homepage URL of the language team.

References:

https://www.gnu.org/software/gettext/manual/html_node/Header-Entry.html

Severity, certainty:

minor, certain

5.8 boilerplate-in-last-translator

The Last-Translator header field contains xgettext boilerplate. It should contain the last translator's name and email address.

References:

https://www.gnu.org/software/gettext/manual/html_node/Header-Entry.html

Severity, certainty:

normal, certain

5.9 boilerplate-in-project-id-version

The Project-Id-Version header field contains xgettext boilerplate. It should contain the name and the version of the package.

References:

https://www.gnu.org/software/gettext/manual/html_node/Header-Entry.html

Severity, certainty:

minor, certain

5.10 boilerplate-in-report-msgid-bugs-to

The Report-Msgid-Bugs-To header field contains xgettext boilerplate. It should contain an email address or URL where one can report bugs in the untranslated strings.

References:

https://www.gnu.org/software/gettext/manual/html_node/Header-Entry.html

Severity, certainty:

normal, certain

5.11 broken-encoding

Header fields and messages contained by this file couldn't be decoded to Unicode. The usual cause of this is incorrect or missing encoding declaration.

Note that in the absence of encoding declaration, i18nspector assumes ASCII encoding.

References:

https://www.gnu.org/software/gettext/manual/html_node/Header-Entry.html#index-encoding-of-PO-files
<https://tools.ietf.org/html/rfc2045#section-5>

Severity, certainty:

serious, possible

5.12 c-format-string-argument-type-mismatch

There's a type mismatch between a C format argument in `msgid` and the corresponding format argument in `msgid_plural`; or between a C format argument in `msgstr` and `msgid`; or between a C format argument in `msgstr[N]` and corresponding `msgid` or `msgid_plural`.

References:

printf(3)

https://www.gnu.org/software/gettext/manual/html_node/Plural-forms.html

Severity, certainty:

serious, possible

5.13 c-format-string-error

A C format string could not be parsed.

References:

printf(3)

Severity, certainty:

serious, possible

5.14 c-format-string-excess-arguments

A C format string for `msgid` consumes more arguments than `msgid_plural`; or `msgstr` consumes more arguments than `msgid`; or `msgstr[N]` consumes more arguments than corresponding `msgid` or `msgid_plural`.

References:

printf(3)

https://www.gnu.org/software/gettext/manual/html_node/Plural-forms.html

Severity, certainty:

serious, possible

5.15 c-format-string-missing-arguments

A C format string for `msgid` consumes fewer arguments than `msgid_plural`; or `msgstr` consumes fewer arguments than `msgid`; or `msgstr[N]` consumes fewer arguments than corresponding `msgid` or `msgid_plural`.

Note that in some languages, the commonly used Plural-Forms expression evaluates to the same value for `n=1` and `n=21`, `n=31`, and so on. Take this Serbian translation for example:

```
Plural-Forms: nplurals=3; plural=n%10==1 && n%100!=11 ? 0 : n%10>=2 && n%10<=4 && (n
↪%100<10 || n%100>=20) ? 1 : 2;
...
msgid "one byte"
msgid_plural "%d bytes"
```

(continues on next page)

(continued from previous page)

```
msgstr[0] "%d bajt"
msgstr[1] "%d bajta"
msgstr[2] "%d bajtova"
```

Here %d should not be replaced with the spelled-out form *jedan*. Either %d should be kept, or the Plural-Forms expression should be amended, so that there is a special case for *n=1*:

```
Plural-Forms: nplurals=4; plural=n==1 ? 3 : n%10==1 && n%100!=11 ? 0 : n%10>=2 && n%10
↳ <=4 && (n%100<10 || n%100>=20) ? 1 : 2
...
msgid "one byte"
msgid_plural "%d bytes"
msgstr[0] "%d bajt"
msgstr[1] "%d bajta"
msgstr[2] "%d bajtova"
msgstr[3] "jedan bajt"
```

References:

printf(3)

https://www.gnu.org/software/gettext/manual/html_node/Translating-plural-forms.html

Severity, certainty:

serious, possible

5.16 c-format-string-non-portable-conversion

A C format string uses a conversion specified or length modifier, for which a more portable replacement exists:

- For integer conversions (%d, %i, %o, %u, %x, and %X), use the `ll` length modifier instead of `L` or `q`.
- For floating-point conversions (%a, %A, %e, %E, %f, %F, %g, and %G), don't use the `l` length modifier.
- Use the `z` length modifier instead of `Z`.
- Use `%lc` instead of `%C`.
- Use `%ls` instead of `%S`.

References:

printf(3)

Severity, certainty:

pedantic, possible

5.17 c-format-string-redundant-flag

A C format string includes a redundant character flag. Either it's a duplicate, or it has no effect:

- The `+` flag overrides the *space* flag.
- The `-` flag overrides the `0` flag.
- If a precision is given, the `0` flag has no effect on integer conversions (%d, %i, %o, %u, %x, and %X).

References:

`printf(3)`

Severity, certainty:

pedantic, possible

5.18 codomain-error-in-plural-forms

Either a plural form value is outside the declared range, or some values within the declared range can never be reached. This normally indicates an error in the plural form expression.

References:

https://www.gnu.org/software/gettext/manual/html_node/Plural-forms.html

Severity, certainty:

serious, certain

5.19 codomain-error-in-unused-plural-forms

Either a plural form value is outside the declared range, or some values within the declared range can never be reached. (But there are no translated messages which use plural forms.) This normally indicates an error in the plural form expression.

References:

https://www.gnu.org/software/gettext/manual/html_node/Plural-forms.html

Severity, certainty:

normal, certain

5.20 conflict-marker-in-header-entry

The header contains a conflict marker (`#-#-#-#-# ... #-#-#-#-#`). The conflict will have to be resolved manually.

References:

https://www.gnu.org/software/gettext/manual/html_node/Creating-Compendia.html#Creating-Compendia

Severity, certainty:

serious, certain

5.21 conflict-marker-in-translation

One of the translated messages appear to contain a conflict marker (`#-#-#-#-# ... #-#-#-#-#`). The conflict will have to be resolved manually.

References:

https://www.gnu.org/software/gettext/manual/html_node/Creating-Compendia.html#Creating-Compendia

Severity, certainty:

serious, possible

5.22 conflicting-message-flags

Two flags with conflicting meanings are associated with one of the messages.

References:

https://www.gnu.org/software/gettext/manual/html_node/PO-Files.html

Severity, certainty:

important, possible

5.23 date-from-future

The date refers to the future. As such, it's extremely unlikely to be correct.

Severity, certainty:

normal, certain

5.24 distant-header-entry

The header entry in this file is preceded by other entries. The header entry should be always the first one.

Severity, certainty:

important, certain

5.25 duplicate-flag-for-header-entry

Multiple identical flags are associated with the header entry.

Severity, certainty:

minor, certain

5.26 duplicate-header-entry

This file contains multiple header entries.

Severity, certainty:

serious, certain

5.27 duplicate-header-field

This file contains multiple header fields of the same name.

Severity, certainty:

minor, wild-guess

5.28 duplicate-header-field-content-transfer-encoding

This file contains multiple Content-Transfer-Encoding header fields.

Severity, certainty:

pedantic, certain

5.29 duplicate-header-field-content-type

This file contains multiple Content-Type header fields.

Severity, certainty:

serious, certain

5.30 duplicate-header-field-date

This file contains multiple date header fields of the same name.

Severity, certainty:

normal, certain

5.31 duplicate-header-field-language

This file contains multiple Language header fields.

Severity, certainty:

important, certain

5.32 duplicate-header-field-language-team

This file contains multiple Language-Team header fields.

Severity, certainty:

normal, certain

5.33 duplicate-header-field-last-translator

This file contains multiple Last-Translator header fields.

Severity, certainty:

normal, certain

5.34 duplicate-header-field-mime-version

This file contains multiple MIME-Version header fields.

Severity, certainty:

pedantic, certain

5.35 duplicate-header-field-plural-forms

This file contains multiple Plural-Forms header fields.

Severity, certainty:

serious, certain

5.36 duplicate-header-field-project-id-version

This file contains multiple Project-Id-Version header fields.

Severity, certainty:

minor, certain

5.37 duplicate-header-field-report-msgid-bugs-to

This file contains multiple Report-Msgid-Bugs-To header fields.

Severity, certainty:

normal, certain

5.38 duplicate-header-field-x-poedit

This file contains multiple X-Poedit-* header fields.

Severity, certainty:

normal, certain

5.39 duplicate-message-definition

This file contains multiple definitions of the same message.

Severity, certainty:

serious, certain

5.40 duplicate-message-flag

Multiple identical flags are associated with one of the messages.

Severity, certainty:

minor, certain

5.41 empty-file

This file doesn't contain any messages.

Severity, certainty:

normal, certain

5.42 empty-msgid-message-with-plural-forms

The message with empty msgid contains plural forms. Such messages are reserved by GNU gettext for header entries, and your code should not call `gettext("", ...)`.

Severity, certainty:

serious, certain

5.43 empty-msgid-message-with-source-code-references

The message with empty msgid contains plural forms. Such messages are reserved by GNU gettext for header entries, and your code should not call `gettext("")`.

Severity, certainty:

serious, possible

5.44 encoding-in-language-header-field

The language header field contains encoding declaration. Such information shouldn't be included in this field.

References:

https://www.gnu.org/software/gettext/manual/html_node/Header-Entry.html

Severity, certainty:

minor, certain

5.45 fuzzy-header-entry

The header entry is marked as fuzzy. For compatibility with very old (<< 0.11) **msgfmt(1)** versions, which didn't support fuzzy header entries, it shouldn't be marked as such.

References:

<https://git.savannah.gnu.org/cgit/gettext.git/tree/NEWS?id=v0.11#n44>

Severity, certainty:

pedantic, certain

5.46 inconsistent-leading-newlines

Some strings in an entry start with a newline, but some don't. Either all of them should start with a newline, or none of them should.

Severity, certainty:

important, possible

5.47 inconsistent-number-of-plural-forms

Number of plural forms in a message definition doesn't match number of plural forms declared in another message definition.

References:

https://www.gnu.org/software/gettext/manual/html_node/Plural-forms.html

https://www.gnu.org/software/gettext/manual/html_node/Translating-plural-forms.html

Severity, certainty:

serious, certain

5.48 inconsistent-trailing-newlines

Some strings in an entry end with a newline, but some don't. Either all of them should end with a newline, or none of them should.

Severity, certainty:

important, possible

5.49 incorrect-number-of-plural-forms

Number of plural forms in a message definition doesn't match number of plural forms declared in the header.

References:

https://www.gnu.org/software/gettext/manual/html_node/Plural-forms.html

https://www.gnu.org/software/gettext/manual/html_node/Translating-plural-forms.html

Severity, certainty:

serious, certain

5.50 invalid-content-transfer-encoding

Value of the Content-Transfer-Encoding header field is invalid. It should be 8bit.

References:

https://www.gnu.org/software/gettext/manual/html_node/Header-Entry.html

<https://tools.ietf.org/html/rfc2045#section-6.1>

Severity, certainty:

pedantic, certain

5.51 invalid-content-type

Value of the Content-Type header field should is invalid. It should be in the form `text/plain; charset=encoding`.

References:

https://www.gnu.org/software/gettext/manual/html_node/Header-Entry.html#index-encoding-of-PO-files

<https://tools.ietf.org/html/rfc2045#section-5>

Severity, certainty:

important, possible

5.52 invalid-date

The date is invalid or in an invalid format. The format should be `YYYY-MM-DD hh:mm+ZZzz`, e.g. `2011-11-05 10:14+0100`.

References:

https://www.gnu.org/software/gettext/manual/html_node/Header-Entry.html

Severity, certainty:

normal, certain

5.53 invalid-language

The Language header field couldn't be parsed, or it contains an unknown language.

References:

https://www.gnu.org/software/gettext/manual/html_node/Header-Entry.html

Severity, certainty:

important, possible

5.54 invalid-language-team

The Language-Team header field contains an e-mail address that uses a reserved domain name, or a partially qualified domain name.

References:

<https://tools.ietf.org/html/rfc2606>

Severity, certainty:

normal, possible

5.55 invalid-last-translator

The Last-Translator header field could neither be parsed as an e-mail, or the e-mail address uses a reserved domain name, or a partially qualified domain name.

References:

https://www.gnu.org/software/gettext/manual/html_node/Header-Entry.html

<https://tools.ietf.org/html/rfc2606>

Severity, certainty:

normal, possible

5.56 invalid-mime-version

Value of the MIME-Version header field is invalid. It should be 1 . 0 .

References:

<https://tools.ietf.org/html/rfc2045#section-4>

Severity, certainty:

pedantic, certain

5.57 invalid-mo-file

This file couldn't be parsed a MO file.

Severity, certainty:

serious, certain

5.58 invalid-range-flag

A `range :` flag couldn't be parsed, or the designated range contained fewer than two numbers. The syntax is `range : min . . max`, where both values are non-negative integers.

References:

https://www.gnu.org/software/gettext/manual/html_node/PO-Files.html

Severity, certainty:

important, certain

5.59 invalid-report-msgid-bugs-to

The `Report-Msgid-Bugs-To` header field could neither be parsed as an e-mail nor as a URL, or the e-mail address uses a reserved domain name, or a partially qualified domain name.

References:

https://www.gnu.org/software/gettext/manual/html_node/Header-Entry.html

<https://tools.ietf.org/html/rfc2606>

Severity, certainty:

normal, possible

5.60 language-disparity

Language of this file has been declared in multiple places, but the declarations don't match.

Severity, certainty:

normal, possible

5.61 language-team-equal-to-last-translator

`Language-Team` and `Last-Translator` header fields contain the same e-mail address.

Severity, certainty:

minor, possible

5.62 language-variant-does-not-affect-translation

The `Language` header field contains a variant designator that is not relevant for the message translation.

References:

https://www.gnu.org/software/gettext/manual/html_node/Header-Entry.html

Severity, certainty:

minor, possible

5.63 leading-junk-in-plural-forms

The Plural-Forms header field contains unexpected text before the `nplurals=` string.

GNU gettext runtime ignores such leading junk, but other header parsers might be less liberal in what they accept.

Severity, certainty:

important, certain

5.64 malformed-xml

The original string or the translated string contains an XML fragment, which is not well-formed.

References:

<https://www.w3.org/TR/REC-xml/#sec-well-formed>

Severity, certainty:

serious, possible

5.65 no-content-transfer-encoding-header-field

The Content-Transfer-Encoding header field doesn't exist. It should be set to `8bit`.

References:

https://www.gnu.org/software/gettext/manual/html_node/Header-Entry.html

<https://tools.ietf.org/html/rfc2045#section-6.1>

Severity, certainty:

pedantic, certain

5.66 no-content-type-header-field

The Content-Type header field doesn't exist. It should be set to `text/plain; charset=encoding`.

Note that in the absence of encoding declaration, i18nspector assumes ASCII encoding.

References:

[https:](https://www.gnu.org/software/gettext/manual/html_node/Header-Entry.html#index-encoding-of-PO-files)

[//www.gnu.org/software/gettext/manual/html_node/Header-Entry.html#index-encoding-of-PO-files](https://www.gnu.org/software/gettext/manual/html_node/Header-Entry.html#index-encoding-of-PO-files)

<https://tools.ietf.org/html/rfc2045#section-5>

Severity, certainty:

important, certain

5.67 no-date-header-field

The date header field doesn't exist.

References:

https://www.gnu.org/software/gettext/manual/html_node/Header-Entry.html

Severity, certainty:

minor, certain

5.68 no-language-header-field

The Language header field doesn't exist.

References:

https://www.gnu.org/software/gettext/manual/html_node/Header-Entry.html

Severity, certainty:

pedantic, certain

5.69 no-language-team-header-field

The Language-Team header field does not exist. It should contain English name of the language, and the email address or homepage URL of the language team.

References:

https://www.gnu.org/software/gettext/manual/html_node/Header-Entry.html

Severity, certainty:

pedantic, certain

5.70 no-last-translator-header-field

The Last-Translator header field doesn't exist. It should contain the last translator's name and email address.

References:

https://www.gnu.org/software/gettext/manual/html_node/Header-Entry.html

Severity, certainty:

normal, certain

5.71 no-mime-version-header-field

The MIME-Version header field doesn't exist. It should be set to 1.0.

References:

<https://tools.ietf.org/html/rfc2045#section-4>

Severity, certainty:

pedantic, certain

5.72 no-package-name-in-project-id-version

The Project-Id-Version header field doesn't appear to contain any name. It should contain both the name and the version of the package.

References:

https://www.gnu.org/software/gettext/manual/html_node/Header-Entry.html

Severity, certainty:

minor, possible

5.73 no-plural-forms-header-field

The Plural-Forms header field does not exist, even though some of the messages use plural forms (although none of them have been translated).

References:

https://www.gnu.org/software/gettext/manual/html_node/Plural-forms.html

Severity, certainty:

minor, certain

5.74 no-project-id-version-header-field

The Project-Id-Version header field does not exist. It should contain the name and the version of the package.

References:

https://www.gnu.org/software/gettext/manual/html_node/Header-Entry.html

Severity, certainty:

minor, certain

5.75 no-report-msgid-bugs-to-header-field

The Report-Msgid-Bugs-To header field does not exist or it is empty. It should contain an email address or URL where one can report bugs in the untranslated strings.

References:

https://www.gnu.org/software/gettext/manual/html_node/Header-Entry.html

Severity, certainty:

normal, certain

5.76 no-required-plural-forms-header-field

The Plural-Forms header field does not exist, even though some of the translated messages use plural forms.

References:

https://www.gnu.org/software/gettext/manual/html_node/Plural-forms.html

Severity, certainty:

serious, certain

5.77 no-version-in-project-id-version

The Project-Id-Version header field doesn't appear to contain any version. It should contain both the name and the version of the package.

References:

https://www.gnu.org/software/gettext/manual/html_node/Header-Entry.html

Severity, certainty:

pedantic, possible

5.78 non-ascii-compatible-encoding

This file uses an encoding that is not compatible with ASCII.

Severity, certainty:

serious, certain

5.79 non-portable-encoding

This file uses an encoding that is not widely supported by software.

References:

https://www.gnu.org/software/gettext/manual/html_node/Header-Entry.html#index-encoding-list

Severity, certainty:

important, certain

5.80 os-error

An input/output error or another operating system error occurred while checking this file.

Severity, certainty:

serious, certain

5.81 partially-translated-message

Translation is missing for some plural forms of a message.

Severity, certainty:

serious, possible

5.82 perl-brace-format-string-error

A Perl format string could not be parsed.

References:

https://www.gnu.org/software/gettext/manual/html_node/perl_002dformat.html

Severity, certainty:

serious, possible

5.83 perl-brace-format-string-missing-argument

A Perl format string for `msgid` doesn't use a named argument that is used in `msgid_plural`; or `msgstr` doesn't use a named argument that is used in `msgid`; or `msgstr[N]` doesn't use a named argument that is used in corresponding `msgid` or `msgid_plural`.

Note that in some languages, the commonly used Plural-Forms expression evaluates to the same value for `n=1` and `n=21`, `n=31`, and so on. Take this Serbian translation for example:

```
Plural-Forms: nplurals=3; plural=n%10==1 && n%100!=11 ? 0 : n%10>=2 && n%10<=4 && (n
↪ %100<10 || n%100>=20) ? 1 : 2;
...
msgid "one byte"
msgid_plural "{n} bytes"
msgstr[0] "{n} bajt"
msgstr[1] "{n} bajta"
msgstr[2] "{n} bajtova"
```

Here `{n}` should not be replaced with the spelled-out form `jedan`. Either `{n}` should be kept, or the Plural-Forms expression should be amended, so that there is a special case for `n=1`:

```
Plural-Forms: nplurals=4; plural=n==1 ? 3 : n%10==1 && n%100!=11 ? 0 : n%10>=2 && n%10
↪ <=4 && (n%100<10 || n%100>=20) ? 1 : 2
...
msgid "one byte"
msgid_plural "{n} bytes"
msgstr[0] "{n} bajt"
msgstr[1] "{n} bajta"
msgstr[2] "{n} bajtova"
msgstr[3] "jedan bajt"
```

References:

https://www.gnu.org/software/gettext/manual/html_node/perl_002dformat.html

https://www.gnu.org/software/gettext/manual/html_node/Translating-plural-forms.html

Severity, certainty:

serious, possible

5.84 perl-brace-format-string-unknown-argument

A Perl format string for `msgid` uses a named argument that isn't used in `msgid_plural`; or `msgstr` uses a named argument that isn't used in `msgid`; or `msgstr[N]` uses a named argument that isn't used in corresponding `msgid` or `msgid_plural`. This indicates that the conversion would try to consume an argument that weren't supplied.

References:

https://www.gnu.org/software/gettext/manual/html_node/perl_002dformat.html

https://www.gnu.org/software/gettext/manual/html_node/Plural-forms.html

Severity, certainty:

serious, possible

5.85 python-brace-format-string-argument-type-mismatch

There's a type mismatch between a Python format argument in `msgid` and the corresponding format argument in `msgid_plural`; or between a Python format argument in `msgstr` and `msgid`; or between a Python format argument in `msgstr[N]` and `msgid` or `msgid_plural`.

References:

<https://docs.python.org/2/library/string.html#formatstrings>

Severity, certainty:

serious, possible

5.86 python-brace-format-string-error

A Python format string could not be parsed.

References:

<https://docs.python.org/2/library/string.html#formatstrings>

Severity, certainty:

serious, possible

5.87 python-brace-format-string-missing-argument

A Python format string for `msgid` doesn't use a named argument that is used in `msgid_plural`; or `msgstr` doesn't use a named argument that is used in `msgid`; or `msgstr[N]` doesn't use a named argument that is used in corresponding `msgid` or `msgid_plural`.

Note that in some languages, the commonly used Plural-Forms expression evaluates to the same value for `n=1` and `n=21`, `n=31`, and so on. Take this Serbian translation for example:

```
Plural-Forms: nplurals=3; plural=n%10==1 && n%100!=11 ? 0 : n%10>=2 && n%10<=4 && (n
↳ %100<10 || n%100>=20) ? 1 : 2;
...
msgid "one byte"
msgid_plural "{n} bytes"
msgstr[0] "{n} bajt"
msgstr[1] "{n} bajta"
msgstr[2] "{n} bajtova"
```

Here {n} should not be replaced with the spelled-out form `jedan`. Either {n} should be kept, or the Plural-Forms expression should be amended, so that there is a special case for `n=1`:

```
Plural-Forms: nplurals=4; plural=n==1 ? 3 : n%10==1 && n%100!=11 ? 0 : n%10>=2 && n%10
↳ <=4 && (n%100<10 || n%100>=20) ? 1 : 2
...
msgid "one byte"
msgid_plural "{n} bytes"
msgstr[0] "{n} bajt"
msgstr[1] "{n} bajta"
msgstr[2] "{n} bajtova"
msgstr[3] "jedan bajt"
```

References:

<https://docs.python.org/2/library/string.html#formatstrings>
https://www.gnu.org/software/gettext/manual/html_node/Translating-plural-forms.html

Severity, certainty:

serious, possible

5.88 python-brace-format-string-unknown-argument

A Python format string for `msgid` uses a named argument that isn't used in `msgid_plural`; or `msgstr` uses a named argument that isn't used in `msgid`; or `msgstr[N]` uses a named argument that isn't used in corresponding `msgid` or `msgid_plural`. This indicates that the conversion would try to consume an argument that weren't supplied.

References:

<https://docs.python.org/2/library/string.html#formatstrings>
https://www.gnu.org/software/gettext/manual/html_node/Plural-forms.html

Severity, certainty:

serious, possible

5.89 python-format-string-argument-number-mismatch

A Python format string for `msgid` consumes more arguments than `msgid_plural`; or `msgstr` consumes more arguments than `msgid`; or `msgstr[N]` consumes more arguments than `msgid` or `msgid_plural`.

Python, unlike C, requires that all unnamed arguments must be consumed during conversion.

References:

<https://docs.python.org/2/library/stdtypes.html#string-formatting-operations>
https://www.gnu.org/software/gettext/manual/html_node/Python.html#Python

Severity, certainty:

serious, possible

5.90 python-format-string-argument-type-mismatch

There's a type mismatch between a Python format argument in `msgid` and the corresponding format argument in `msgid_plural`; or between a Python format argument in `msgstr` and `msgid`; or between a Python format argument in `msgstr[N]` and `msgid` or `msgid_plural`.

References:

<https://docs.python.org/2/library/stdtypes.html#string-formatting-operations>

Severity, certainty:

serious, possible

5.91 python-format-string-error

A Python format string could not be parsed.

References:

<https://docs.python.org/2/library/stdtypes.html#string-formatting-operations>

Severity, certainty:

serious, possible

5.92 python-format-string-missing-argument

A Python format string for `msgid` doesn't use a named argument that is used in `msgid_plural`; or `msgstr` doesn't use a named argument that is used in `msgid`; or `msgstr[N]` doesn't use a named argument that is used in corresponding `msgid` or `msgid_plural`.

Note that in some languages, the commonly used Plural-Forms expression evaluates to the same value for `n=1` and `n=21`, `n=31`, and so on. Take this Serbian translation for example:

```
Plural-Forms: nplurals=3; plural=n%10==1 && n%100!=11 ? 0 : n%10>=2 && n%10<=4 && (n
↪ %100<10 || n%100>=20) ? 1 : 2;
...
msgid "one byte"
msgid_plural "%(n)d bytes"
msgstr[0] "%(n)d bajt"
msgstr[1] "%(n)d bajta"
msgstr[2] "%(n)d bajtova"
```

Here `%d` should not be replaced with the spelled-out form `jedan`. Either `%d` should be kept, or the Plural-Forms expression should be amended, so that there is a special case for `n=1`:

```

Plural-Forms: nplurals=4; plural=n==1 ? 3 : n%10==1 && n%100!=11 ? 0 : n%10>=2 && n%10
↪ <=4 && (n%100<10 || n%100>=20) ? 1 : 2
...
msgid "one byte"
msgid_plural "%(n)d bytes"
msgstr[0] "%(n)d bajt"
msgstr[1] "%(n)d bajta"
msgstr[2] "%(n)d bajtova"
msgstr[3] "jedan bajt"

```

References:

<https://docs.python.org/2/library/stdtypes.html#string-formatting-operations>
https://www.gnu.org/software/gettext/manual/html_node/Translating-plural-forms.html

Severity, certainty:

serious, possible

5.93 python-format-string-multiple-unnamed-arguments

A Python format string uses multiple unnamed arguments (such as %d). The translator might need to reorder the arguments to properly translate the message, but this is not possible with unnamed arguments. Named arguments (such as %(num)d) should be used instead.

References:

<https://docs.python.org/2/library/stdtypes.html#string-formatting-operations>
https://www.gnu.org/software/gettext/manual/html_node/Python.html#Python

Severity, certainty:

serious, possible

5.94 python-format-string-obsolete-conversion

A Python format string uses an obsolete conversion specifier:

- Use %d instead of %u.

References:

<https://docs.python.org/2/library/stdtypes.html#string-formatting-operations>

Severity, certainty:

pedantic, possible

5.95 python-format-string-redundant-flag

A Python format string includes a redundant character flag. Either it's a duplicate, or it has no effect:

- The + flag overrides the *space* flag.
- The - flag overrides the 0 flag.

- If a precision is given, the 0 flag has no effect on integer conversions (%d, %i, %o, %u, %x, and %X).

References:

<https://docs.python.org/2/library/stdtypes.html#string-formatting-operations>

Severity, certainty:

pedantic, possible

5.96 python-format-string-redundant-length

A Python format string includes a redundant length modifier. Length modifiers (h, l, or L) have no effect in Python.

References:

<https://docs.python.org/2/library/stdtypes.html#string-formatting-operations>

Severity, certainty:

pedantic, possible

5.97 python-format-string-redundant-precision

A C format string includes precision that has no effect on the conversion.

References:

<https://docs.python.org/2/library/stdtypes.html#string-formatting-operations>

Severity, certainty:

pedantic, possible

5.98 python-format-string-unknown-argument

A Python format string for `msgid` uses a named argument that isn't used in `msgid_plural`; or `msgstr` uses a named argument that isn't used in `msgid`; or `msgstr[N]` uses a named argument that isn't used in corresponding `msgid` or `msgid_plural`. This indicates that the conversion would try to consume an argument that weren't supplied.

References:

<https://docs.python.org/2/library/stdtypes.html#string-formatting-operations>

https://www.gnu.org/software/gettext/manual/html_node/Plural-forms.html

Severity, certainty:

serious, possible

5.99 python-format-string-unnamed-plural-argument

A Python format string uses an unnamed arguments (such as %d) in the context of plural forms. The translator might want not to use the numeric argument in the singular form; but this is not possible if the argument is unnamed, because

Python, unlike C, requires that all unnamed arguments must be consumed during conversion. Named arguments (such as `% (n) d`) should be used instead.

References:

<https://docs.python.org/2/library/stdtypes.html#string-formatting-operations>
https://www.gnu.org/software/gettext/manual/html_node/Python.html#Python

Severity, certainty:

wishlist, possible

5.100 qt-plural-format-mistaken-for-c-format

A `c-format` flag is associated with the message, but `qt-plural-format` should be used instead.

The only C format directive that the message uses is `%n`. It is very atypical to use it alone in a C format string.

References:

<https://doc.qt.io/qt-5/i18n-source-translation.html#handling-plurals>
`printf(3)`

Severity, certainty:

important, possible

5.101 range-flag-without-plural-string

A `range:` flag is associated with a message that doesn't have plural string. `range:` flags only make sense for translations involving plural forms.

References:

https://www.gnu.org/software/gettext/manual/html_node/PO-Files.html

Severity, certainty:

normal, certain

5.102 redundant-message-flag

A flag associated with one of the messages is redundant, because it's implied by another flag.

References:

https://www.gnu.org/software/gettext/manual/html_node/PO-Files.html

Severity, certainty:

pedantic, certain

5.103 stray-header-line

The header contains a line that does not belong to any header field. Note that RFC-822-style folding of long headers is not supported.

References:

<https://lists.gnu.org/archive/html/bug-gettext/2012-12/msg00010.html>

Severity, certainty:

important, certain

5.104 stray-previous-msgid

The message entry contains annotations about previous untranslated string (`#| msgid...`), even though the message is not marked as fuzzy. These annotations are only useful for fuzzy messages, and should be removed when unfuzzifying.

References:

https://www.gnu.org/software/gettext/manual/html_node/PO-Files.html

Severity, certainty:

minor, certain

5.105 syntax-error-in-plural-forms

Value of the Plural-Forms header field could not be parsed. It should be in the form `nplurals=n;` `plural=expression.`

References:

https://www.gnu.org/software/gettext/manual/html_node/Plural-forms.html

Severity, certainty:

serious, certain

5.106 syntax-error-in-po-file

This file couldn't be parsed a PO file. In some rare cases this is due to incorrect or missing encoding declaration.

References:

https://www.gnu.org/software/gettext/manual/html_node/PO-Files.html

<https://bugs.debian.org/692283>

Severity, certainty:

serious, possible

5.107 syntax-error-in-unused-plural-forms

Value of the Plural-Forms header field could not be parsed. (But there are no translated messages which use plural forms.) It should be in the form `nplurals=n; plural=expression.`

References:

https://www.gnu.org/software/gettext/manual/html_node/Plural-forms.html

Severity, certainty:

important, certain

5.108 trailing-junk-in-plural-forms

The Plural-Forms header field contains unexpected text after the plural expression.

GNU gettext runtime ignores such trailing junk, but other header parsers might be less liberal in what they accept.

Severity, certainty:

important, certain

5.109 translation-in-template

The PO template file contains a translated message.

Severity, certainty:

minor, certain

5.110 unable-to-determine-language

i18nspector was unable to determine language of this file. Absence of this information will prevent it from performing further checks.

Severity, certainty:

normal, wild-guess

5.111 unexpected-flag-for-header-entry

An unexpected flag is associated with the header entry. The only flag that makes sense for the header entry is `fuzzy`.

References:

https://www.gnu.org/software/gettext/manual/html_node/PO-Files.html

Severity, certainty:

normal, possible

5.112 unknown-encoding

This file declares an encoding that couldn't be recognized by i18nspector. It might be a typo. Absence of encoding information will prevent i18nspector from performing further checks.

Severity, certainty:

important, possible

5.113 unknown-file-type

File format of this file couldn't be recognized. It might be a bug in i18nspector.

Severity, certainty:

normal, wild-guess

5.114 unknown-header-field

The header field name is unknown to i18nspector. It might be a typo or a capitalization error (header field names are case-sensitive).

References:

<https://lists.gnu.org/archive/html/bug-gettext/2012-12/msg00010.html>

Severity, certainty:

minor, wild-guess

5.115 unknown-message-flag

An unknown flag is associated with one of the messages. It might be a typo.

References:

https://www.gnu.org/software/gettext/manual/html_node/PO-Files.html

Severity, certainty:

normal, wild-guess

5.116 unknown-poedit-language

Language declared in X-Poedit-Language couldn't be recognized. It might be a bug in i18nspector.

Severity, certainty:

minor, wild-guess

5.117 unrepresentable-characters

The declared encoding cannot represent all characters commonly used in this language. This is a strong indication that the declared encoding is incorrect.

Severity, certainty:

serious, possible

5.118 unusual-character-in-header-entry

The header entry contains an unusual character. This is usually an indication of an encoding problem, such as:

- using ISO 2022 escape sequences, or
- using UTF-8 despite declaring an 8-bit encoding.

References:

https://www.unicode.org/faq/utf_bom.html#bom6

Severity, certainty:

important, certain

5.119 unusual-character-in-translation

One of the translated messages contains an unusual character. This is usually an indication of an encoding problem, such as:

- using ISO 2022 escape sequences, or
- using UTF-8 despite declaring an 8-bit encoding.

References:

https://www.unicode.org/faq/utf_bom.html#bom6

Severity, certainty:

important, possible

5.120 unusual-plural-forms

The Plural-Forms declaration is incorrect (or unusual), according to i18nspector's linguistic data.

Severity, certainty:

serious, possible

5.121 unusual-unused-plural-forms

The Plural-Forms declaration is incorrect (or unusual), according to i18nspector's linguistic data. (But there are no translated messages which use plural forms.)

Severity, certainty:

normal, possible

CHAPTER 6

See also

msgfmt(1), particularly the `-c` option