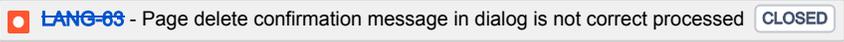


# I18n Translations vs. MessageFormat

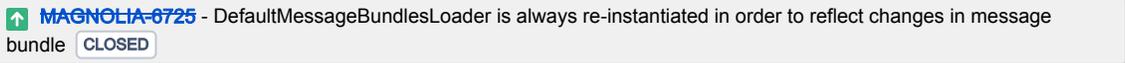
## Problems

1. Usage of MessageFormat in i18n translations is inconsistent (as of 5.5.0)
2. Developers cannot figure when to use MessageFormat syntax, and when not to

- Translations fetched "fluently" from definitions are not message-formatted.
- Translations fetched explicitly via `SimpleTranslator / i18n#translate` are processed by MessageFormat.
  - Few of these translations mandate argument formatting or pluralization
  - Many of these translations are still expected to be plain text
- Should developers face escaping of single quotes (as double-single quotes)?
  -  **LANG-63** - Page delete confirmation message in dialog is not correct processed CLOSED
- Should developers draw special attention to angle-brackets?
  - MessageFormat vs. our alleged HTML sanitation
  - Currently angle brackets need an extra space not to be considered as HTML tags
  - Alleviated by browser concatenating white-space, but the extra-space remains irrelevant.
    - should translations be re-used in system messages, logs, or native apps to name a few
  - To be further elaborated upon

## Proposals

General idea: Make it explicit which keys go into MessageFormat, and which don't.

-  **Prefix/suffix for keys**
    - `foo.bar.label.formatted = @MessageFormat {0} élément(s) supprimés.`
  - Wrap value into e.g. braces?
    - `foo.bar.label = ``{0} élément(s) supprimés.```
  - Prefix value with annotation?
    - `foo.bar.label = @MessageFormat {0} élément(s) supprimés.`
  -  **Work on SimpleTranslator side**
    - `request foo.bar.label`
      - => look up for `foo.bar.label.formatted`
      - => fall back on `foo.bar.label`
    - No impact on Java code, invocations of `SimpleTranslator, i18n#translate`
    - Can spare message-formatting "plain" values
    - One more lookup
      - ⚠ otoh lookup might be suboptimal atm
      - triggering refresh? see  **MAGNOLIA-6725** - defaultMessageBundlesLoader is always re-instantiated in order to reflect changes in message bundle CLOSED
  - treated as a separate issue
-  MessageFormat must be used consistently, i.e. leaving escaping of single-quotes to developer/translator
  - Roman: we tried to doubled the quotes in the code before formatting and failed (because there is too many use cases)
-  **Not upgrade-friendly, but the sooner the better**
  - devs' existing keys (unsuffixed) using MessageFormat would stop working
  - A. 5.5 is young enough to go for big fat release notes
  - B. we *could* detect presence of MessageFormat syntax and dump some (read lots of) deprecation warnings
  - C. still use MessageFormat for "plain" keys, along with a deprecation warning
    - tentatively avoiding detection—could be applied only when arguments are passed along
-  **Meanwhile, we double the single-quotes in our own formatted translations anyway**

See also



SUPPORT-5445 - Jira project doesn't exist or you don't have permission to view it.



MGNLUI-3701 - IllegalArgumentException is thrown when is formatted invalid message pattern

CLOSED



LANG-42 - Correct i18n messages to be valid for formatting using MessageFormat

CLOSED



MGNLUI-3746 - Use MessageFormatterUtils instead MessageFormat for message formatting

OPEN

—should we really "swallow" MessageFormat exceptions?

indeed, users shouldn't face formatting exceptions; it's enough for them to see the message is visually borked; but we probably want to digest it on a higher level, rather than inflicting our util upon all usages?