TYPO3.Flow - Feature # 7724

Status:	Resolved	**Priority:**	Must have
Author:	Karol Gusak	**Category:**	I18n
Created:	2010-05-09	**Assigned To:**	Karol Gusak
Updated:	2010-10-20	**Due date:**	
PHP Version:			
Has patch:			
Complexity:			
Subject:	Implement Message Formatter subsystem		

Description

Message Formatter is a class which formats a string given by substituting placeholders with properly formatted values. Placeholders consists of an unsigned integer enclosed in curly braces. Optionally format type can be set, and this one can set even more precisely by format style. Examples:

- You have {0} items in cart, total value: {1}
- You have {0,number} items in cart, total value: {1,number}
- You have {0,number,integer} items in cart, total value: {1,number,currency}

Format types are classes implementing simple API, so developers can create custom format types, and they can have custom format styles.

Number, Date, and Time format types will be implemented by default.

ViewHelpers for Fluid will be implemented in order to make use of Message Formatter subsystem in template files.

Related issues:

related to TYPO3.Flow - Feature # 6724: Internationalization, locale, multi-l...	**Resolved**

Associated revisions

Revision 4f3e0d41 - 2010-07-02 11:15 - Karol Gusak

[+FEATURE] FLOW3 (Locale): Added base code for message formatting. Relates to #7724.

Revision f27a0afd - 2010-07-02 13:06 - Karol Gusak

[~TASK][~API] FLOW3 (Locale): Renamed DateTimeFormatter to DatetimeFormatter in order to access it as "datetime" (and not "dateTime") in placeholders. Relates to #7724.

Revision 419cba15 - 2010-07-27 17:23 - Karol Gusak

[~TASK] FLOW3 (I18n): Moved formatting functionality from DatesReader to DatetimeFormatter. Relates to #7722, 7724.

Revision b5c6c641 - 2010-07-28 10:50 - Karol Gusak

[~TASK] FLOW3 (I18n): Added new test for DatetimeFormatter. Relates to #7724.

Revision 3a28df25 - 2010-07-28 12:56 - Karol Gusak

[~TASK] FLOW3 (I18n): Moved formatting functionality from NumbersReader to NumberFormatter. Relates to #7722, #7724.
[~TASK] FLOW3 (I18n): Minor code cleaning.

History

#1 - 2010-07-01 16:53 - Karol Gusak

- Status changed from New to Accepted

#2 - 2010-07-02 14:05 - Karol Gusak

- Status changed from Accepted to Needs Feedback
- % Done changed from 0 to 30

Base implementation of message formatting was commited. This is how it works:

- *FormatResolver* is a class which is used to replace all placeholders in string with corresponding formatted values
- It uses classes implementing *FormatterInterface* (i.e. formatters) in order to format a value as defined in the placeholder
- Some formatters use readers (classes named like *FooReader* from *\F3\FLOW3\Locale\Cldr\Reader* namespace)
- Readers won't be accessed directly by application developer. They abstract CLDR and provide methods to operate on this data. For example, *NumberFormatter* uses *NumbersReader* to do actual formatting (as only *NumbersReader* knows the rules/patterns from CLDR)

Please note the naming: *NumbersReader* because it abstracts "numbers" tag from CLDR, but *NumberFormatter* because to use it one should write "number" in placeholder.
Some placeholder examples to make it clear:

- {0}, {1} etc - Places string-casted version of a value (number in brackets defines the value's index from values array provided)
- {0,number,decimal} or {0,number} - Format as decimal number using pattern from CLDR defined for current locale
- {0,datetime} - Format as date and time
- {0,datetime,time,full} - Format only time part, choose the "full" version of pattern from CLDR

Second element in the placeholder defines name of concrete formatter. Remaining elements are passed directly to the formatter (for example, *NumberFormatter* uses only one additional element, defining if format value as decimal, or percentage, but *DatetimeFormatter* accepts two elements - style (datetime, date or time) and length (full, long, medium, short).

After writing this I actually think that *DatetimeFormatter* should be divided into *DateFormatter*, *TimeFormatter*, and *DatetimeFormatter*, which would make defining placeholders more clear (but will be redundant).

And another related topic I would like to ask you about - how should the currency amounts be represented in FLOW3? I think simple *Currency* object would be nice, it would store a decimal number and a international currency symbol.

What do you think?

#3 - 2010-08-15 22:14 - Karol Gusak

- Status changed from Needs Feedback to Resolved
- % Done changed from 30 to 100

This issue is done, related issue opened #9313.

#4 - 2010-08-18 10:48 - Karsten Dambekalns

- Target version set to 1.0 alpha 11