DITA 1.3 specialization: Learning Content Education (LCE)

Documentation for version 3.0

Date: 20 January 2021

Published by XStructuring

[Introduction 2](#_Toc62031924)

[LCE interaction types 3](#_Toc62031925)

[Specialized elements used in LCE interactions types: 4](#_Toc62031926)

[lceChainMatching 4](#_Toc62031927)

[lceDrawing 4](#_Toc62031928)

[lceGapMatch 4](#_Toc62031929)

[lceGraphicAssociate 4](#_Toc62031930)

[lceGraphicGapMatch 5](#_Toc62031931)

[lceGraphicTextEntry 5](#_Toc62031932)

[lceHotspot 5](#_Toc62031933)

[lceHottext 6](#_Toc62031934)

[lceInlineChoice 6](#_Toc62031935)

[lceMultipleMatch 6](#_Toc62031936)

[lcePerformQuestion 6](#_Toc62031937)

[lceAcrosticPuzzle and lceCrossWordPuzzle 7](#_Toc62031938)

[lceTextEntry 7](#_Toc62031939)

[General LCE 8](#_Toc62031940)

[Elements 8](#_Toc62031941)

[Attributes 8](#_Toc62031942)

[Extra information 10](#_Toc62031943)

[Content model within acrostic and crossword 10](#_Toc62031944)

# Introduction

* The LCE specialization is an extension of the DITA 1.3 L&T in which some more interaction types, some domain elements and attributes are defined to make DITA more suitable for educational content.
* The LCE specialization uses elements form standard DITA 1.3 L&T
* The LCE version 3.0 now is called “**L**earning **C**ontent **E**ducation” and not “Learning Content Extra” anymore, because it’s mainly for educational material/publishers and it contains more than only extra interactions.
* LCE 3.0 is based on LCE 2.x. But there is one not downwards compatible change. In LCE 2.x the element lceFixedEntry was introduced with which the content of a gap could be given in advance. This is replaced by the new value “fixed” for the attribute @answertype
* The git repository with the LCE specialization has been moved. The new repository is: <https://github.com/XStructuring/eu.xstructuring.dita.lce>

# LCE interaction types

(in alphabetical order)

* **lceChainMatching** – Kind of extension of the lcMatching2 interaction. In stead of two buckets you can define three or more item buckets. Items between bucket 1 and 2 or bucket 2 and 3 etc. have a 1-to-1 relation.
* **lceDrawing** – Kind of open question. The student has to draw something on a given figure.
* **lceGapMatch** – Text with gaps; drag-and-drop words into the gap. Gaps can have the same answer. More words can be given than needed.
* **lceGraphicAssociate** – The student must determine which hotspots on the image have a relation with each other. Quite often the output shows lines between the related hotspots.
* **lceGraphicGapMatch** – The student has to place words or icons with drag-and-drop onto a figure. The same answer can be used several times. There can be more answers than needed.
* **lceGraphicTextEntry** – A combination of lceGraphicGapMatch and lceTextEntry. The student has to fill in gaps on an image. The gaps are defined the same way as the drop areas in the lceGraphicGapMatch.
* **lceHotspot** – An extension of the lcHotspot to allow some more LCE block elements in the content model
* **lceHottext** – Text in which some interactive words or characters can be highlighted in a web-application through clicking on it.
* **lceInlineChoice** – Text with gaps; for each gap the answer can be selected in a drop down menu.
* **lceMultipleMatch** – Matrix; create pairs from two sets of items, each item can be used several times (n-to-n relation).
* **lcePerformQuestion** – Kind of open question. An offline action and/or upload is expected of the student. There is no answer field and also it is not possible to provide an answer.
* **lceAcrosticPuzzle** – One-dimensional crossword; for each clue a solution word must be filled in in the grid. It is possible to provide an overall solution word build of letters from different puzzle words.
* **lceCrossWordPuzzle** – two-dimensional crossword for each clue a solution / word must be entered in the grid. It is possible to provide an overall solution / word composed of letters from different puzzle words.
* **lceTextEntry** – Fill in the gap; for each gap one correct answer must be given. It is also possible to define one or more alternative answers. Usually an alternative answer gets less score points.

The question types lceTextEntry, lceInlineChoice, lceGapMatch and lceHottext are so called “in-line questions”. Not because the question is standing in-line but the answers are part of a text and thus in-line.

## 

# Specialized elements used in LCE interactions types:

## lceChainMatching

* lceChainTable – Container for the “buckets” with items. A column in the table is a “bucket”.
* lceChainRow – Container for items which have a relation whereby the item in column 1 and 2 have a relation, in column 2 and 3 etc.
* lcItem2 – Standard DITA L&T element. Item in a bucket, this can be text or image.

## lceDrawing

* lceDrawingImage – Container for a figure onto which the student has to draw.
* lceDrawingAnswer – The possible answer, the answer must be checked manually. The answer can be the correct figure and/or some text.

## lceGapMatch

* lceInlineInteractionText – Container for text with gaps.
* lceGap – Gap into which a given word must be placed by drag-and-drop. The answers are listed separately from text with gaps. Each gap refers to an answer using the attribute @href. Thus, one answer can be used in different gaps. In a gap you can give 'lceFeedbackInlineCorrect' and 'lceFeedbackInlineIncorrect'.
* lceGapMap – Container for the possible answers (word or characters) the student has to use.
* lceGapAnswer – Answer, attribute @id is required.

## lceGraphicAssociate

* lceGAMap – Container for the image and the hotspots.
* lceGAMap/image – Image with hotspots
* lceAssociableHotspot – Hotspot, defining shape and coordinates
* lcAreaShape2 – In this element the shape of the drop area is defined. In DITA 1.3 L&T the shapes 'rect', 'circle' and 'poly' are supported.
* lcAreaCoords2 – Contains the coordinates of a drop area.
* lceMatchMap – Container for the pairs of hotspots which have a relation.
* lceMatchPair – Container for one pair of hotspots which have an relation.
* lceAssociable – Reference element to an associable hotspot, use the href attribute here for.

## lceGraphicGapMatch

* lceGGMMap – Container for the image and the answer to be overlaid onto the image by drag-and- drop.
* lceGGMMap/image – Image onto which the answers must be dropped. For the answers drop areas are shown on the image.
* lceAreaGGM – Container for the answer and drop area, its shape and coordinates
* lceAreaContent – Answer, can be text or an image. If after the element ‘lceAreaContent’ no element ‘lcCorrectResponse2’ follows, the answer is just a distractor. No shape and coordinates have to be added in that case.
* lcAreaShape2 – In this element the shape of the drop area is defined. In DITA 1.3 L&T the shapes 'rect', 'circle' and 'poly' are supported.
* lcAreaCoords2 – Contains the coordinates of a drop area

## lceGraphicTextEntry

A mix of graphic gap, match and text entry. As the ‘lceGraphicGapMatch’ the ‘lceGraphicTextentry’ uses some standard DITA 1.3 L&T elements.

* lceGTEMap – Container for the image and the answers which have to be filled out on the image.
* lceGTEMap/image – Image onto which the answers must be filled out. For the answers drop areas are shown on the image.
* lceAreaGTE – Container for the answer and drop area, its shape and coordinates.
* lceAnswerOptionInline – Answer, also see: lceTextEntry
* lcAreaShape2 – In this element the shape of the drop area is defined. In DITA 1.3 L&T the shapes 'rect', 'circle' and 'poly' are supported.
* lcAreaCoords2 – Contains the coordinates of a drop area.

## lceHotspot

In this question type the standard DITA 1.3 L&T elements of lcHotspot2 are used.

* lcHotspotMap2 – Container for the image and hotspots.
* lcHotspotMap2/image – Image onto which the hotspots are placed.
* lcArea2 - Container for hotspot areas, their shape and coordinates.
* lcAreaShape2 – In this element the shape of the drop area is defined. In DITA 1.3 L&T the shapes 'rect', 'circle' and 'poly' are supported.
* lcAreaCoords2 – Contains the coordinates of a drop area.

## lceHottext

* lceInlineInteractionText – Container for the text with interactive words or characters.
* lceHottextOption – Answer, interactive word or character. Correct answers must be marked with 'lcCorrectResponse2'. If 'lcCorrectResponse2' is missing, the word or character is interactive but not a correct answer. Each interactive word can have a 'lceFeedbackInlineCorrect' and 'lceFeedbackInlineIncorrect'.

## lceInlineChoice

* lceInlineInteractionText – Container for the text with gaps and their drop down menus.
* lceInlineChoiceOptions – Container for the drop down menu with the answers. With 'lceFeedbackInlineCorrect' and 'lceFeedbackInlineIncorrect' an individual feedback for one choice group can be given.

lceChoiceOptionInline – Answer. In the element 'lceInlineChoiceOptions' more than one 'lceChoiceOptionInline' must be given otherwise no choise has been made. One answer must be marked as correct answer with 'lcCorrectResponse2’. Child-elements: 'lceAnswerContentInline'.

## lceMultipleMatch

* lceMultipleMatch – This question contains three child-elements: 'lceMatchSet1', 'lceMatchSet2' and 'lceMatchMap'.
* lceMatchSet1 – Container for one or more items which belongs to the first set.
* lceMatchSet2 – Container for one or more items which belongs to the second set.
* lceAssociableItem – Item in 'lceMatchSet1'. Each 'lceAssociableItem' has an attribute 'number' which should contain a unique number/string.
* lceAssociableIMatch – Item in 'lceMatchSet2' which can match with an item from the first set. Each 'lceAssociableMatch' has an attribute 'number' which should contain an unique number/string.
* lceMatchMap – Container for pairs.
* lceMatchPair – Container for a pair. A pair contains an associable item and match from each set. You can build as many pairs as you want. It is allowed to use an item or match in different pairs. If an item or match is not part of a pair than it is an extra word to confuse the student.
* lceAssociable – Reference element to an associable item or match, use the attribute @href.

Note: If you have only direct pairs and all items are used only once, it is recommended to use the DITA 1.3 L&T 'lcMatching2.

## lcePerformQuestion

* lceUploadObject – In the publication an upload button should be generated for this element

## lceAcrosticPuzzle and lceCrossWordPuzzle

* lcePuzzleGrid – Grid in which an acrostic or crossword should be build. You can fill in the puzzle letters in the desired grid entries. The element ‘lceCrossWordGrid’ is based on a table and should be shown as a table in the editor.
* lcePuzzleRow – A row in the puzzle grid.
* lcePuzzleEntry – A grid cell (table cell) in which a letter can be added. All filled ‘lcePuzzleEntries’ must have an id.
* lcePuzzleOptionGroup – Container for all puzzle options.
* lcePuzzleOption – Container for one clue and its matching puzzle word. Puzzle options should automatically be numbered or so.
* lcePuzzleClue – Clue about the puzzle word.
* lcePuzzleAnswer – Answer/puzzle word. The answer consists of references ‘lcePuzzleEntryRef’ to letters in the grid. Furthermore in the answer a 'lceFeedbackInlineCorrect' and 'lceFeedbackInlineIncorrect' can be given.
* lcePuzzleEntryRef – Reference to a letter in the grid which belongs to this special clue answer. The reference to the first letter, thus the first reference in the answer, should be the trigger to generate a number or other signs in the grid cell where the words starts so that the student can see which clue belongs here.
* lcePuzzleSolution – A special paragraph for the overall solution of a puzzle.
* lcePuzzleSolutionWord – The overall solution. As in a clue answer the solution word contains references to letters in the grid. Because of these references the letters in the grid can get special numbers or colors so that the students know that these letters are important for the overall solution.

Note: For details around answers and linking to letters in the grid, see below.

## lceTextEntry

* lceInlineInteractionText – Container for the text with gaps.
* lceTextEntryOptions – Container for the answer(s); beside the correct answer alternative answers can be given which are partly correct.
* lceAnswerOptionInline – Answer. Child-elements 'lceAnswerContentInline', 'lcCorrectResponse2', lceAlternativeResponse', 'lceFeedInlineCorrect' and 'lceFeedInlineIncorrect'. After the element 'lceAnswerContentInline' must stand the element 'lcCorrectResponse2' or 'lceAlternativeResponse' to define if the answer is correct or just an alternative (nearly correct).
* lceAnswerContentInline – The answer for a gap like a word, number or character.
* 'lceFeedbackInline'. The content model concerning feedback is similar to the one in lcSingleSelect2.
* lceAnswerContentInline – A word, number or character.

# General LCE

## Elements

* lceBox – A specialization of fig. This can be used for special information which should get a special lay-out, usually as framed box. The lceBox element has the same display attributes as fig. Also you can add a title and description to a box. If you want to place a figure or table in the lceBox element you first have to insert a div element.
* lceBR – An empty element with no attributes. The same as the br element in HTML: line break
* lceExplanation – This element is allowed in all DITA L&T interaction 2 elements beside lceHotspot2 (therefor the lceHotspot has been declared) and all LCE interaction elements. The lceExplanation is a specialization of fig and stands in the redefine of the xs:group for lcAsset2. In all lce graphic interactions the lceExplanation is added directly into the content models because they do not allow lcAsset2. Thus lceExplanation stands in the xml after the lcQuestion2 element. The lceExplanation is a special element for extra information for the student after he/she has filled out the question and need some explanation for the answer. This explanation is not a kind of feedback, it is neutral and independent of whether the question has been answered correctly or incorrectly. Even in the XML the lceExplanation stands after the lcQuestion2 element is digital applications, it just should be shown on demand.
* lceFeedbackInline, lceFeedbackInlineCorrect, lceFeedbackInlineIncorrect – Feedback element based on element ph and can be used in specialized elements which are based also on element ph or p.
* lceEduData – Container element for educational information, should only be used in interactions even if it is referenced in the xs:group data and thus allowed everywhere, where data is allowed.
* lceLomEdu – Container element for the set of metadata elements which are defined for lom educational e.g. lomDifficulty or lomTypicalLearningTime.
* lceRefpoint – An empty element which only has an id and should be used as reference target in text if you cannot reference to a standard element.

The elements ‘score’ and ‘lceEduData’ (and children) can also be used in DITA 1.3 (L&T) interaction elements.

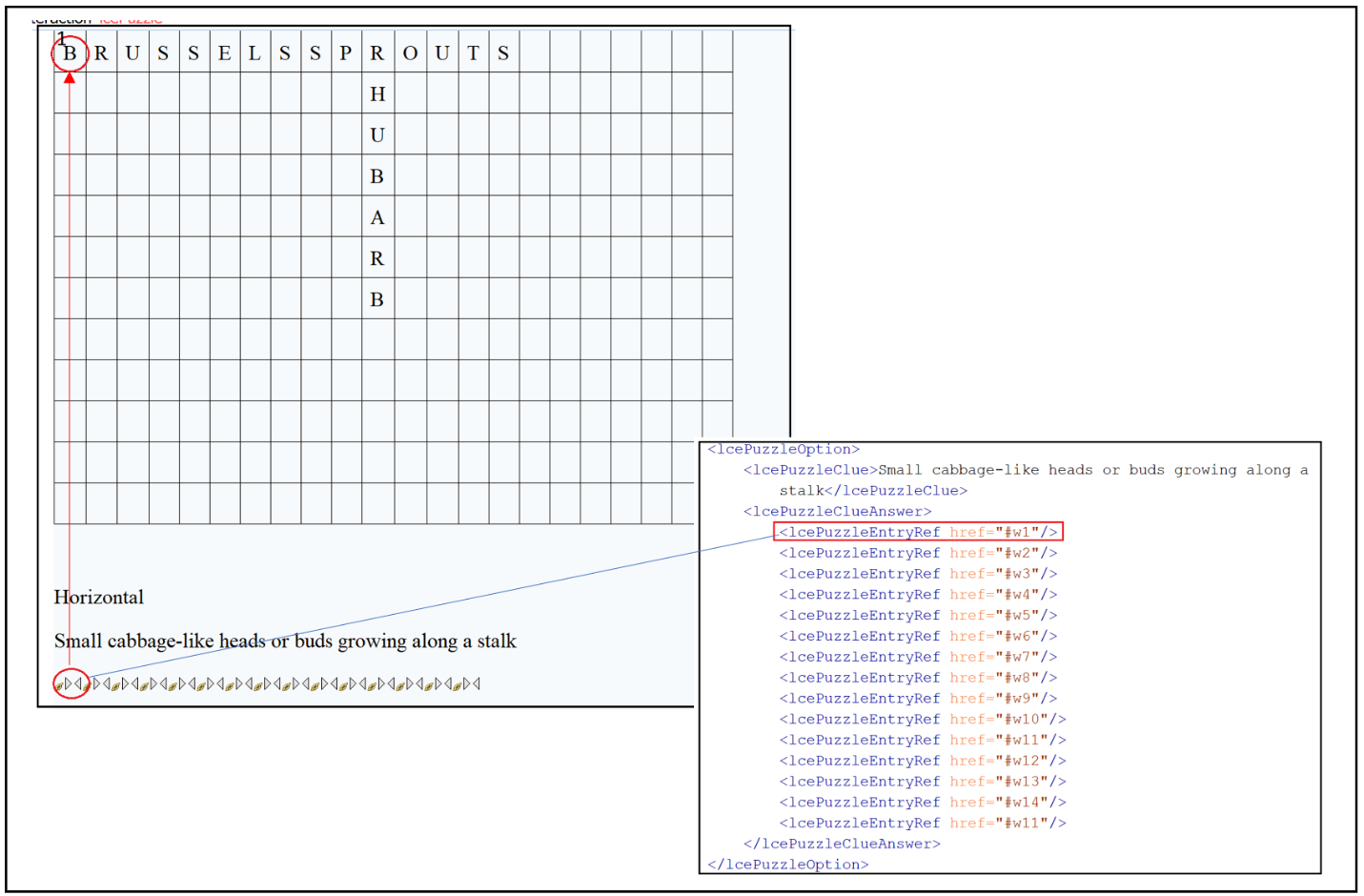
## Attributes

* answertype – values: predefined or fixed. This attribute should only be used with answer elements[[1]](#footnote-1). A predefined answer is an example of how an answer should look like; a fixed answer is not really an answer but text in place of the answer. This is especially defined for the lceGraphicTextEntry but can also be used within other interactions.
* answerformat – values: text or numerical. The answer format gives the application an indication which kind/format the answer has, especially within gaps.
* answerlength – values: xs, s, m, l, xl. With this attribute the expected length of an answer can be given e.g. the expected length of a gap within an lceTextEntry.
* case-sensitive – values: yes or no. In a text entry interaction it can be important if the answer is case sensitive or not. This attribute can also be used in puzzle interactions.
* contentclass – open text field. In the contentclass the semantic meaning of element content can be given like poem, summary, instruction or assignment. Quite often the contentclass attribute value will be also used for the output layout but it should never contain pure layout information. And anyway, the semantic meaning of some content and its layout can differ e.g.<lceBox contentclass=”grammar” outputclass=”post-it”>.
* orientation – values: horizontal or vertical. Can be used in a crossword with the ‘lcePuzzleOption’ to specify the orientation of the word in the grid. This attribute can also be used in sequence interaction.
* placeholdertext – open text field. In this attribute text can be stored which will be shown to the student and must be overwritten like a processing instruction in XML.
* score – open text field. Can be used for an interaction of single answers to define the score. The score is needed when questions are part of an assessment that will be judged.
* shuffle – values: yes or no. If you want to define if answers must be shuffled or not. Usually this is done by the publication application but with this attribute you can overrule it.

All attributes can also be used for DITA 1.3 (L&T) elements.

# Extra information

## Content model within acrostic and crossword



1. See below. [↑](#footnote-ref-1)