ISSN 0965-5425, Computational Mathematics and Mathematical Physics, 2013, Vol. 53, No. 3, pp. 369–374. © Pleiades Publishing, Ltd., 2013.
Original Russian Text © S.A. Abramov, E.A. Bordachenkova, D.E. Khmel'nov, 2013, published in Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki, 2013, Vol. 53, No. 3, pp. 495–501.

Extendable Essay as a Hypertext Scheme for Information and Educational Material

S. A. Abramov[a], E. A. Bordachenkova[b], and D. E. Khmel'nov[a]

[a] Dorodnicyn Computing Center, Russian Academy of Sciences, ul. Vavilova 40, Moscow, 119333 Russia
[b] Faculty of Computational Mathematics and Cybernetics, Moscow State University,
Leninskie gory, Moscow, 119992 Russia
e-mail: sergeyabramov@mail.ru, lenabord@mail.ru, dennis_khmelnov@mail.ru

Received August 7, 2012; in final form, September 24, 2012

Abstract—A hypertext scheme for an extendable essay, i.e., a brief presentation of educational, scientific, informative, or reference material with supplementary, bibliographic, illustrative and other optional extensions is proposed. A particular example of an extendable essay is described. The differences between extendable essays and available hypertext schemes are discussed. An implementation of the scheme that makes it possible to create extendable essays of any content is described.

DOI: 10.1134/S0965542513030020

Keywords: extendable essay, hypertext, subordinated extensions, XML hypertext markup language, tag, XSL stylesheet language.

1. INTRODUCTION

According to an encyclopedic dictionary [1], "an essay (from French essai meaning "trial" or "attempt") is a genre ... combining the author's own views ... with an informal presentation involving spoken language." The history of the essay genre and the term itself was traced back (in a genre of essay!) in [2]. Essay is a published genre. In oral genres, it can be compared to a close-to-monologue conversation in which the more informed interlocutor, who can be referred to as an expert, outlines any subject, for example, a new scientific direction. The listener occasionally asks for explanations, examples, etc. Such a brief narrative with requested supplements, which are referred to as *extensions*, can provide a simple and easy-to-understand introduction to any scientific or practical area, which is useful for gaining a deeper insight into the subject in the future.

In our work, we attempt to use a hypertext approach to combine thoughtfully selected, tested, and prepared material with the flexibility of an oral presentation. According to Wikipedia [3], "... hypertext is a set of texts involving references (hyperlinks) to each other that allow the reader to choose information to be read or a sequence of reading" (other similar definitions of hypertext can be found, for example, in [4]). Below we consider a hypertext scheme (form, principle) for an *extendable essay*, i.e., a brief presentation of educational, scientific, informative, or reference material with supplementary, bibliographic, illustrative and other optional extensions; extensions containing test questions to be answered after reading are also possible.

Initially, an extendable essay looks like a rather short text displayed on a screen that provides an introductory overview of the subject. If a paragraph arousing interest is labeled on the margin by an extension mark of one or another type, it can be extended. The resulting extended text can also be extended, etc. The reader can reject any extension and return to the previous version of the text. The final version can be printed out.

The extensions proposed can be called *soft*, since they increase the amount of information gradually without sharp jumps. In this context, they differ from, for example, articles in Wikipedia, where the transition to a more detailed text leads to a sharp increase in information and requires more intensive focus on the text. However, the main difference is that, after an extension, a Wikipedia article is no more a unified text. Meanwhile, a specific feature of extendable essays is that an initially linear text remains such after all extensions and rejections. Here, "linear" is meant not figuratively, so that differently placed text pieces can be read one after another in a chosen sequence, but in a common sense. The resulting text remains within the essay genre. It does not involve any seams or other secondary consequences of the extensions performed. The structure of the material, its logic, and presentation style remain unchanged, which, hopefully, enables the reader to rather easily comprehend the sequential stages of the text transformation. Thus, the scheme proposed facilitates easy comprehension of texts.

A traditional hypertext is well understood if the reader is generally familiar with the subject area. In the authors' view, thanks to its linearity, the structure of an extendable essay is more suitable for an introductory presentation of material. Note another feature of an extendable essay that is useful for a beginner. When the reader wants to grasp the idea of a theoretical or practical domain, it happens that the depth of his or her immersion into material is not known beforehand; it depends on how the material will be understood by the reader. An extendable essay makes it possible to stop when enough material has been overviewed.

An important concept for an extendable essay is the *subordination* of extensions. If the material appearing in an extension has to be preceded by another extension skipped by the reader, the latter is opened automatically and is marked in a special manner (for more detail, see Section 2.1). This is associated with the problem of preserving the focus of reading, when an automatically opened segment lies above in the text. This problem is solved in our implementation (see Section 3.3).

The types of extensions proposed can be somehow used, for example, in university courses. However, such courses have other aims: the material to be presented is determined by the curriculum and the point is not to provide the student with somewhat less or more material, but to present this material in a comprehensible form to facilitate the grasp of a large amount of knowledge. In this case, an extendable essay can serve as an introductory survey of a lecture course, while providing self-contained material (an example is given in Section 2 below).

Most of the opportunities provided by an extendable essay can be found in some form in available hypertext systems. For example, some electronic dictionaries provide an option to extend the list of synonyms, many learning systems offer test questions, etc. However, to the authors' knowledge, the entire scheme proposed, which models an introductory expert consultation, has not been previously considered. In Section 3, we describe an implementation of this scheme in its general form as based on the XML/XSL technology [5, 6]: a set of XML tags and the corresponding transformation file ExtEss.xsl (Extendable Essay) are proposed. It allows one to prepare an extendable essay with any content.

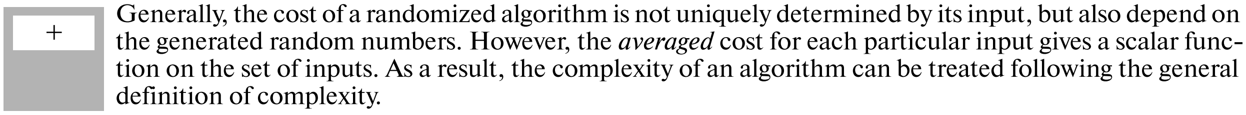
Instead of creating a new XML document structure, the extendable essay scheme can be implemented as an extension of some available XML document structure or a new module in an existing system. For example, the proposed scheme can be added to the SCORM standard [7] or can be implemented as an extension module in a learning environment, for example, in Moodle [8]. However, in this case, much effort has to be spent on the incorporation of the scheme into existing structures and systems and on taking into account the existing rules and constraints with the result being a mixture of old and new opportunities. For this reason, we have preferred to implement an individual XML document structure, which clearly illustrates all the ideas of the scheme and can be used as a reference model for incorporation into other structures and systems.

2. POSSIBLE SOFT EXTENSIONS

2.1. Control of the Amount of Material

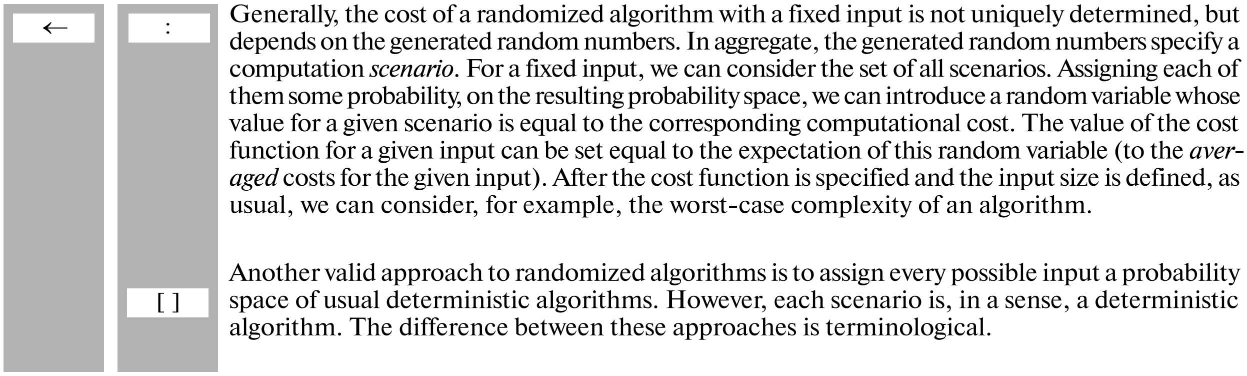
As a trial version, we have created an extendable essay "Complexity of Algorithms" based on a lecture course [9]. The initial text of the essay consists of 11 brief sections (items), each containing one to four paragraphs. The types of extensions used in the essay are described below.

For example, item 5 of the essay discusses the complexity of randomized algorithms. It consists of only three paragraphs, the second of which is as follows:



Generally, the cost of a randomized algorithm is not uniquely determined by its input, but also depend on the generated random numbers. However, the *averaged* cost for each particular input gives a scalar function on the set of inputs. As a result, the complexity of an algorithm can be treated following the general definition of complexity.

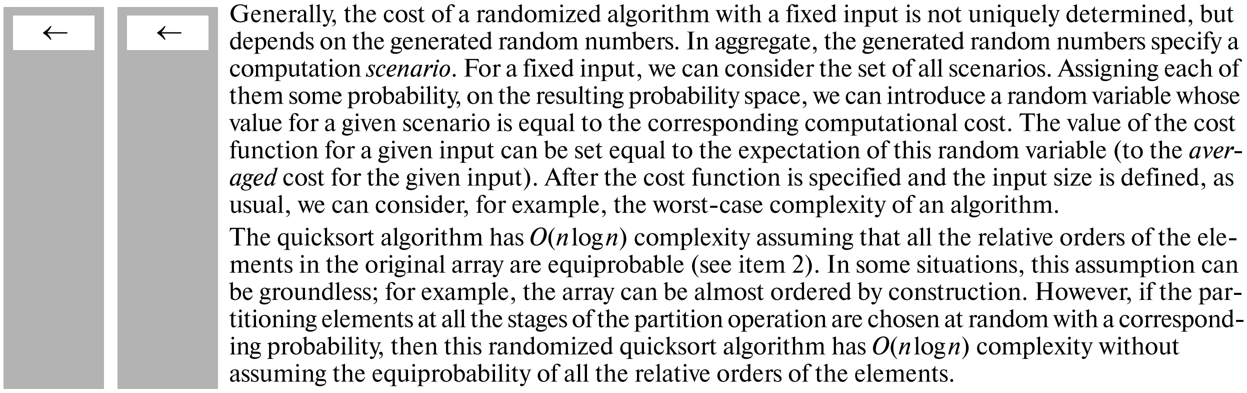
The + sign on the margin says that this text has an extension (when the extension is opened, the + sign is replaced by a folding sign ←).



Generally, the cost of a randomized algorithm with a fixed input is not uniquely determined, but depends on the generated random numbers. In aggregate, the generated random numbers specify a computation *scenario*. For a fixed input, we can consider the set of all scenarios. Assigning each of them some probability, on the resulting probability space, we can introduce a random variable whose value for a given scenario is equal to the corresponding computational cost. The value of the cost function for a given input can be set equal to the expectation of this random variable (to the *averaged* costs for the given input). After the cost function is specified and the input size is defined, as usual, we can consider, for example, the worst-case complexity of an algorithm.

Another valid approach to randomized algorithms is to assign every possible input a probability space of usual deterministic algorithms. However, each scenario is, in a sense, a deterministic algorithm. The difference between these approaches is terminological.

This extended version (referred to hereafter as the first extension) has two paragraphs. The : sign on the margin of the first extension offers the opportunity of adding an illustrative example.

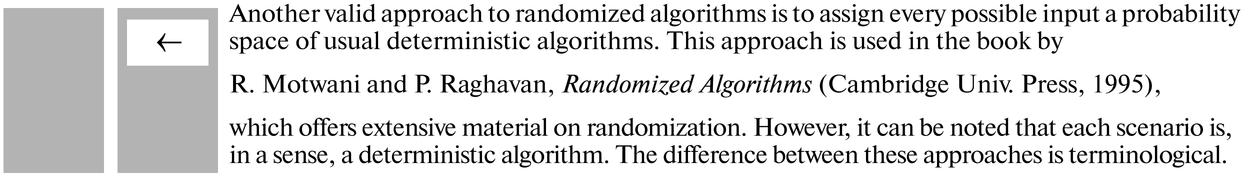


Generally, the cost of a randomized algorithm with a fixed input is not uniquely determined, but depends on the generated random numbers. In aggregate, the generated random numbers specify a computation *scenario*. For a fixed input, we can consider the set of all scenarios. Assigning each of them some probability, on the resulting probability space, we can introduce a random variable whose value for a given scenario is equal to the corresponding computational cost. The value of the cost function for a given input can be set equal to the expectation of this random variable (to the *averaged* cost for the given input). After the cost function is specified and the input size is defined, as usual, we can consider, for example, the worst-case complexity of an algorithm.

The quicksort algorithm has $O(n\log n)$ complexity assuming that all the relative orders of the elements in the original array are equiprobable (see item 2). In some situations, this assumption can be groundless; for example, the array can be almost ordered by construction. However, if the partitioning elements at all the stages of the partition operation are chosen at random with a corresponding probability, then this randomized quicksort algorithm has $O(n\log n)$ complexity without assuming the equiprobability of all the relative orders of the elements.

The last paragraph contains a reference to item 2 of the essay, where quicksort is mentioned in one of the extensions. The partition of the original version of the essay into numbered items is useful for such references. However, quicksort is discussed not in the original version of item 2, but in the extension of one of its paragraphs. If that extension was not opened by the reader while he or she was reading item 2, it will be opened automatically after opening the discussed extension (automatic extension concerns item 2, while the reader stays in item 5 and nothing changes on the screen; whether or not to look at item 2 is up to the reader). A *subordinated* extension is one requiring that another (*leading*) extension be opened, since it uses information presented in the latter. The folding sign for an automatically opened leading extension is ⇐. It visually differs from ←, which helps the reader quickly find these extensions in the text.

Now we consider the second paragraph of the first extension. The [] sign on the margin says that there is a bibliographic reference.



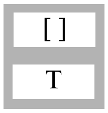
Another valid approach to randomized algorithms is to assign every possible input a probability space of usual deterministic algorithms. This approach is used in the book by

R. Motwani and P. Raghavan, *Randomized Algorithms* (Cambridge Univ. Press, 1995),

which offers extensive material on randomization. However, it can be noted that each scenario is, in a sense, a deterministic algorithm. The difference between these approaches is terminological.

The last two extensions are *nested*: they serve as extensions of paragraphs arising after opening an extension at an earlier stage, in this case, the first extension.

Item 5 is ended with two empty paragraphs:



An extension whose sign is on the margin of an empty paragraph corresponds to the entire item of the text. On the margin of the first empty paragraph, there is a [ ] sign. This means a reference to [9], more exactly, to Section 8 of this book, which discusses the averaged cost and the complexity of randomized algorithms:

← S. A. Abramov, *Lectures on the Complexity of Algorithms* (MTsNMO, Moscow, 2012) (Section 8).

2.2. Sets of Test Questions

The margin of the second (last) empty paragraph in item 5 contains a T sign (“training”, “testing”), which means that there is an option to receive a test question concerning the entire material presented in item 5. In each such situation, a question is chosen randomly from a set of questions available in the given item. The reader cannot see the entire set. Several answer options are offered, and the reader has to mark one as true. For example, after item 5 has been read, the following question can be offered:

← *Question*: Is it true that the determination of the averaged cost of a randomized algorithm requires specifying a probability distribution:

1. on the set of all scenarios (see above in this item) assigned to each particular admissible input?
2. on each of the sets of all inputs of fixed size?

Choose the answer: ▾

1
2

If the reader chooses 1, the response is

+ 1 is the correct answer.

In the other cases, the response is

+ The correct answer is 1.

If a question assumes acquaintance with a certain extension of one of the preceding paragraphs and this extension was not executed, it is opened automatically when the question appears on the screen (the first extension in our case).

The feedback obtained by the reader makes it possible to execute an extension signed by +, which provides a brief explanation of the correct answer.

← The correct answer is 1:
To determine the averaged cost of a randomized algorithm A with an input x, one needs the probabilities of the scenarios assigned by A to x. Then the computational cost for each such scenario define a random variable on the probability space of scenarios assigned to x. Its expectation is the value of the averaged cost function for x.

2.3. List of Extensions

Summarizing, the following extensions and their signs are used in an essay:

+—expand in more detail,
:—give an example,
[ ]—give a reference,
T—ask a test question and check the answer.

The first three signs denote extensions that replace a paragraph by a larger text (sometimes, a single paragraph is replaced by several ones). Formally, a single sign could be used for all three types of extensions. However, the signs employed tell the reader about the character of possible extensions and facilitate the understanding of an extended text. They help decide whether to open an extension of this type or skip it.

The sign T corresponds to a special type of extension. The reader's answer to a test question is received and checked.

Recall that, as soon as an extension is opened, its sign on the margin is replaced by a folding sign ←, which makes it possible to return to the unextended paragraph, if desired. If an extension is opened automatically (see Sections 2.1, 2.2), the folding sign has a slightly different form: ⇐. It should be noted that the folding of a leading extension is blocked as long as at least one of its subordinated extensions is open. As a result, the text on the screen is always a full-fledged essay.

3. IMPLEMENTATION OF THE EXTENDABLE ESSAY SCHEME

3.1. XML/XSL Technology

The implementation of an extendable essay is underlain by the XML technology [5], which allows the creation of structured data in text format with the structure specified with the help of special text marks—tags. For our implementation, we have defined a set of tags with which the structure and the content of an essay, namely, the basic text, extensions (including nested ones), and test questions can be written in a natural manner. All these tags are described below.

An XML text can be conversed to another text format with the help of an XSL transformation [6]. This transformation is represented as a text file with a set of instructions in the XSLT language.

We have created an XSL file that transforms the XML file of an essay into an HTML file. The latter can then be opened in any modern browser (for example, Internet Explorer, Safari, or Chrome). The resulting HTML file contains the necessary elements that open and fold extensions, including subordinated ones, show test questions, and check the answers. The basic difficulty in the implementation lies in the XSL transformation, which contains the logic of the essay structure and the rules for the generation and insertion of control elements and mechanisms for processing of actions with these control elements.

Data in XML format have a tree structure, i.e., the basic part of an element of data between the opening and closing tags (for brevity, simply an element with the name of the corresponding tag) can include other elements, which in turn can also have nested elements. The tree structure, together with the tags, determines the semantics of an XML text. In the case of our implementation, this structure ensures deriving, with the help of the XSL transformation, a correctly working HTML file of an extendable essay.

3.2. Basic Tags of the Extendable Essay Scheme

The root of the XML tree of an extendable essay is an element **Essay**. Inside **Essay**, there can be an element **Title**, which defines the title of the essay, and one or several elements **Part**. With the help of an element **Part**, the essay is divided into consecutive parts (in Section 2, they were called items). Inside each element **Part**, there can be an element **Number**, which specifies the index of this part, and one or several elements **Paragraph**, which correspond to the paragraphs of the given part of the essay.

Inside each element **Paragraph**, there can be an element **Basic**, which specifies the basic text of the given paragraph, and an element **Extension**, which defines an extension of the given paragraph. The attribute **type** of an element **Extension** defines the extension type: "**More**", "**Cite**," and "**Example**" for extensions with additional information, a bibliographic reference, or an example, respectively. Inside each element **Extension**, in turn, there can be one or several elements **Paragraph**, which define the paragraphs of the given extension and nested extensions, if some of the elements **Paragraph**, in addition to **Basic**, includes an element **Extension**.

Specified by the corresponding tags, these elements determine the basic structure of an essay, which is a consecutive set of textual parts consisting of paragraphs, which can have extensions, in turn, consisting of paragraphs, possibly, with nested extensions.

Inside an element **Part**, there can be an element **Tasks**, which determines a set of test questions concerning the given part of the essay. Inside an element **Tasks**, there can be one or several elements **Task**, which specify these questions.

Inside an element **Task**, there must be an element **Question** (statement of a test question), an element **Options** (a set of elements **Option** specifies the answer variants for the given question), an element **Answer** (the index of the correct answer among the variants inside the element **Options**), and an element **Comment** (explanations of the correct answer).

When opening a T extension, the reader receives a test question chosen randomly from the set defined inside the element **Tasks**.

3.3. Subordinated Extensions

Subordinated extensions (the text of a subordinated extension requires opening the corresponding leading extension, since the former uses information presented in the latter) are implemented using the mechanism of labels. A paragraph whose extension may be entailed by an extension of another paragraph is assigned a label. For this purpose, the corresponding element **Paragraph** is supplemented with an attribute **label**, whose value is used to specify subordination. The paragraph label is used in a subordinated extension. For this purpose, the element **Extension** is supplemented with an additional attribute **linked** with a value corresponding to the label of that paragraph whose extension has to be opened together with the given subordinated extension.

In addition to extensions of paragraphs, subordination can be attributed to test questions, for example, if the statement of a question or the explanation of a correct answer uses information presented only in a certain extension. In this case, an element **Task** can receive an additional attribute **linked** with the value corresponding to the label of the paragraph whose extension has to be opened if the given test question is chosen.

An extension that is leading for another one can be subordinated to a certain extension. In this case, all the subordinated extensions are opened in a sequence. It may also happen that a certain leading extension is nested. In this case, the surrounding extensions are also opened.

3.4. Additional Tags

Several additional tags are available for simple formatting of texts inside paragraphs:

em indicates that the text becomes italic;

bf indicates that the text becomes bold;

p indicates that the text begins with a new line;

center indicates that the text is centered;

indent indicates that the text becomes indented (several successive tags indicates a double, triple, etc., indent);

img allows inserting an image in the text; the image is contained in a file with the path specified by the **src** attribute; in this way, images of complex formulas that cannot be written as simple text can be inserted in the text (images of formulas have to be prepared in advance).

3.5. Extendable Essay on Implementation

A more detailed description of the implementation of an extendable essay has been created in the form of a document with the help of the technique described above, i.e., in the form an extendable essay. Its original XML text can be used as a sample essay, since it contains examples of all the possibilities described above. The latest versions of this document in XML and HTML format and of the transformation XSL file ExtEss.xsl, as well as the extendable essay “Complexity of Algorithms,” can be found at http://www.ccas.ru/ca/doku.php/essay.

REFERENCES

1. *Soviet Encyclopedic Dictionary* (Sovetskaya Entsiklopediya, Moscow, 1985) [in Russian].
2. A. K. Zholkovsky, “Essay,” in *Beware of the Tripod!* (Vremya, Moscow, 2010) [in Russian].
3. http://ru.wikipedia.org/wiki/
4. http://slovar.lib.ru/dictionary/gipertext.htm
5. http://www.w3.org/TR/xml11/
6. http://www.w3.org/Style/XSL/
7. *A Review of SCORM 2004* (Informika, Moscow, 2005).
8. A. Kh. Gil’mutdinov, R. A. Ibragimov, and I. V. Tsivil’skii, *E-Learning Based on Moodle* (Kazan Gos. Univ., Kazan, 2008).
9. S. A. Abramov, *Lectures on the Complexity of Algorithms* (MTsNMO, Moscow, 2012) [in Russian].

Translated by I. Ruzanova