

MarkHub Cloud Online Editor as a modern web-based book creation tool

Vitalina O. Babenko¹[0000-0002-4816-4579], Roman M. Yatsenko²[0000-0001-7968-6890],
Pavel D. Migunov¹ and Abdel-Badeeh M. Salem³[0000-0001-7710-4842]

¹ V.N. Karazin Kharkiv National University, 4 Svobody Sq., Kharkiv, 61022, Ukraine
vitalinababenko@karazin.ua, xa12462349@student.karazin.ua

² Simon Kuznets Kharkiv National University of Economics,
9-a Nauky Ave., Kharkiv, 61166, Ukraine
roman.yatsenko@hneu.net

³ Ain Shams University, El-Khalyfa El-Mamoun Str., Cairo, Egypt
abmsalem@yahoo.com

Abstract. The main criterion for the competitiveness of a teacher or expert in the field of science is a good ability to present their knowledge to students in an interactive form without spending a lot of time in preparation. The purpose of the study is to analyze modern editors to create educational information content in the modern educational space and to present a modern tool for creating web books based on the latest IT technologies. Modern editors of web material creation have been analyzed, statistics of situations on mastering of knowledge by listeners, using interactive methods of information submission have been investigated. Using the WYSIWYG concept and analyzing modern information tools for presenting graphic material, an effective tool for teaching interactive web material was presented. An adapted version of the MarkHub online editor based on cloud technologies is presented. Using MarkHub cloud-based online editor for the unified development of educational content can significantly increase the author's productivity in the content creation process. At the same time, the effects of reducing the time spent on formatting the external presentation of the content, making synchronous changes to different versions of the content, tracking the versions of the content, organizing remote teamwork in the network environment are achieved.

Keywords: web-book, interactive material, WYSIWYG, WYSIWYM, cloud-based online editor MarkHub, book modeling.

1 Introduction

In the modern educational space, new technologies are being created every day, and they are the ones that give the pace of social development. There are many challenges that need to be addressed quickly to prevent large-scale consequences. This is influenced by several factors: the surrounding atmosphere, the system of education and the demand of young people for learning [15]. Not so long ago, when preparing for

lectures, the teacher used printed sources – mostly textbooks and other literature from scientific libraries. He wrote the main topics he needed to convey to the students, and then gave lectures. More often, this process took a lot of personal time. This has significantly reduced the time needed to improve their professional skills. But in recent times the preparation of lecture material on the basis of printed sources alone is insufficient. The generation of students changes over time, they become more progressive and have more demand for information. Generation Z mostly refers to the “digital person” [5]. Therefore, the lecturer needs to increase the level of creativity in creating quality content for young people. They are used to visualized content, such as on social networks. Theoretical material should be supplemented with presentations and interactive assignments that help you discover and learn practical skills. Studies show that students’ achievement in course development is improved by 38% using interactive video instruction, and learning time is reduced by 31% [10]. This trend is observed not only in students. An average employee in a firm that is advanced training using modern methods of studying material, demonstrates a high degree of mastering the course (more than 80%) [1]. So now, the requirement of interactivity in teaching is added, through which the teacher can impart knowledge through the latest information technologies. Therefore, being competitive in the present is very difficult and, at the same time, it is important to be successful in the professional field. But to reach a certain level, it is necessary to use information tools that will not only reduce the time, for example, to prepare lectures, but also make them interesting.

2 Approaches to the formation of educational content

Modern approaches and technologies used in the preparation of content in the educational environment, require a lot of time and effort of the author. In preparing educational content (guidelines, textbooks, e-courses), the author often devotes most of his time not to the production of new knowledge, but to their visual design. Dissemination of different forms and mediums of content delivery, as a rule, entails the use of different software tools to create appropriate visual versions of the same educational content. At the same time, insufficient attention is paid to the application of methods and models in risk management, taking into account the particularities of the implementation of IT projects, in particular the methodologies used in the creation of software. In this regard, an urgent scientific and applied challenge is to improve the risk management processes of information technology projects by developing appropriate methods and models that combine the benefits of different methodological approaches to software development.

Many of these tools use the concept of user interface What You See Is What You Get (WYSIWYG) [4]. To implement this UI concept, content creation tools use the visual formatting tools of the document closest to the output, using a mouse and predefined keyboard shortcuts. Because in this concept, working on the external representation of content occurs simultaneously with its creation, this entails the following negative consequences:

- working on the external presentation of content distracts the author from his content side, thereby reducing the author's productivity and quality of the creative process;
- content transformation when necessary to form another external representation of it, which often leads to changes in its content and the appearance of new versions;
- versioning the external presentation of content with differences in content negates the quality of the first versions, which often leads to the need to make substantially the same changes in different versions.

The WYSIWYG Content Generation Approach was first introduced in 1974 in a Bravo text editor developed by the Xerox PARC Research Center. In the future, this approach became widespread due to the active introduction of a graphical user interface [14], which actively used visual methods of presenting information and inputting information using the mouse and keyboard manipulator. By the end of the twentieth century, this approach had become the de facto standard when working with content in an educational environment [7]. The most common content preparation package with a similar interface is the Microsoft Office application suite, which includes text, spreadsheet, and presentation applications that are both screen and printable. Moreover, software supporting the WYSIWYG concept, both for different purposes and for different manufacturers, requires the use of individual approaches to create and change the appearance of a document.

Unlike WYSIWIG, the What You See Is What You Mean approach (WYSIWYM) implies the use of content preparation methods in which the logical structure and content of the document are separated from its appearance [8]. This is accomplished by applying different design templates and styles to one original document content. In this way, content independence from form is achieved, which gives the following advantages when creating educational content [11]:

- creating different forms of visual representation of content without the need to make changes to the content or minimize their volume;
- making changes to the contents of a document occurs once without the need for duplication of changes in different versions of the visual representation;
- at the same time, making changes to the layout in one version of the visual representation does not affect the content of the document and its other visual representations;
- increases the ability to automatically analyze and convert content by using text storage formats.

The WYSIWYM concept appeared historically before WYSIWYG, but initially required the use of a sophisticated tag model of document content markup (similar to modern HTML and XML markup languages). In the academic world, the typesetting system TeX (1978, by Donald Knuth [9]) has become widespread, which contains advanced tools for document structuring, typography, link creation, work with complex mathematical formulas, etc. Many large scientific publishers use the modern LaTeX package, which is an extension of TeX. The original content in LaTeX documents can be compared to a computer program because it is a text file containing special markup commands. The use of predefined macros allows to automate many content creation

tasks without involving complex programming skills, which does not remove the need for initial preparation and training for the author before applying this package.

The next step in the development of the WYSIWYM concept is the emergence of lightweight markup languages (Markdown Language, 2004, by John Gruber), which have taken advantage of the separation of content and form but use a much simpler model of content markup. Lightweight markup languages feature the use of a simple set of special characters to denote various logical blocks of content in such a way that its visual perception is as it is, without the involvement of additional tools, which in turn are only necessary to convert the format of the external content representation.

The rapid development and dissemination of the lightweight markup language family that has emerged in the coming years makes it possible to consider the use of the WYSIWYM approach as a basis for forming the concept of unified development of educational content. The proposed concept implies a technology for creating and disseminating educational content based on the WYSIWYM concept, which will meet the following requirements:

- support of various formats of external presentation of content;
- realization of creation of different types of content;
- lack of significant and specific requirements for the level of technical competence of content authors;
- the use of tools with a friendly user interface;
- providing version control and joint remote development of content;
- extensibility by embedding complex visual elements of content, including those created in WYSIWYG editors.

3 Analyzing tools for creating educational content

Educational content has undergone several stages of its evolution and the most common of these are oral knowledge transfer, manuscripts, book printing, e-books. Educational systems in some countries already use e-books in training. It is first and foremost accessibility and mobility that helps students to continuously learn. But nowadays, interactive e-books have more advantages than static e-books. The student has the opportunity to interact with different types of content: videos, pictures, tests, presentations, audio files and not interrupt the learning process. All items are in one online e-book.

Let's analyze the advantages and disadvantages of the most common WYSIWYM editors used in the educational space (Table 1, 2).

Table 1. Editors Characteristics.

Editor	Advantages	Disadvantages
<i>Sublime Text</i> is a text editor for code, markup and prose that supports Python plugins [3]	<ul style="list-style-type: none"> — Nice and easy interface. — Ability to display from 1 to 4 panels, thus you can display 4 files at a time. 	<ul style="list-style-type: none"> — No graphical interface to create color schemes. — No clickability of links.

Editor	Advantages	Disadvantages
	<ul style="list-style-type: none"> – Automatic command completion. (During the process of code-typing, the editor recognizes the programming language and proposes to finish the command, as when writing a SMS to a friend, you are already given a ready-made version of the word). – There is an auto-save feature so that a person does not lose their job in an emergency. – Dark background of the editor allows to increase the contrast of the text. 	<ul style="list-style-type: none"> – It is not possible to add interactive material.
<p><i>Notepad++</i> is an open source text editor for Windows with support for a large number of markup languages [16]</p>	<ul style="list-style-type: none"> – Syntax highlighting. – The ability to auto-complete the code. – Work with many tabs at one time. – Small file size and easy to install. – Shortcuts: create, paste, delete, copy and so on. 	<ul style="list-style-type: none"> – Does not auto-indent after colon. – There is no syntax check. – Does not show docstring users (ability to comment on algorithm or command). – It is not possible to add interactive material.
<p><i>Vim</i> is a text editor, one of the most powerful editors with the freedom to customize and automate through advanced add-ons [17]</p>	<ul style="list-style-type: none"> – Ability to work with multiple documents simultaneously; – Auto-completion. – Unicode support. – History of words, commands. – Ability to compare two files and transfer data from one to another. 	<ul style="list-style-type: none"> – The modal user interface seems strange to people who are used to Windows user interfaces; – The habits of working with this editor (commands, scripts, syntax) will not be with you, if you switch to another editor, they will remain in Vim; – In the Vim editor, writing training takes a lot of time and most people work slower than other editors; – It is not possible to add interactive material.
<p><i>Emacs</i> is a family of extensible text editors [18]</p>	<ul style="list-style-type: none"> – Functionality of the editor: basic and additional modes. – Each function or variable can have its own documentation, like a database, and if you click on it you can see the contents. – Ability to customize editing parameters. – Modifying editor features using Emacs language. 	<ul style="list-style-type: none"> – Slow loading process. – Emacs is not easy to learn. – This editor is in some sense an entire operating system, so some users run because of the lack of bits or the need for a regular and simple editor. – It is not possible to add interactive material.

Editor	Advantages	Disadvantages
<i>LyX</i> is a word processor that helps you focus on the structure and content of your documents instead of constantly being distracted by their formatting [13]	<ul style="list-style-type: none"> — Provides complete control over margins, headers / footers, spacing / indentation, alignment, types of multi-level list markers. — There is a well-worked spreadsheet editor, Emacs-style version control interface for collaborative projects, a change tracking system, and “branching” for parallel versions of documents. — You can increase the fonts on the screen according to your tastes and see all the text on the screen without affecting the margins and other formatting of your final result. — You can export the document in various formats. 	<ul style="list-style-type: none"> — The set of paragraph styles is limited to exactly those styles that are specified in the original document class. — Very low prevalence of .lyx format. Not one package except LyX opens. — It is advisable to know LaTeX to get started with LyX, as many functions may overlap. — There is no way to add interactive material. — You will not be able to make a space between paragraphs as the editor cleans it up, so you will first need to choose a style that suits you according to the text structure.
<i>LaTeX</i> is the most popular set of macro extensions of the TeX computer layout system, which facilitates the collection of complex documents [6]	<ul style="list-style-type: none"> — A large number of macropackages. — Supports any language within a single document. — Mathematical formulas are extremely simple to type. — The user needs to know only a few easy-to-remember commands that define the logical structure of a document, and almost no knowledge of how the document is formatted. 	<ul style="list-style-type: none"> — Simultaneous training and use are not supported, which makes the approach to the development of technology more serious. — No WISIWIG. It is necessary to have the skills to work in a text editor. — Requires knowledge of elementary basics of typesetting. — Focuses on multitasking environments. — It is not possible to add interactive material.

Table 2. Editors Comparison.

	Sublime Text	Note-pad++	Vim	Emacs	LyX	LaTeX	MarkHub	MS Word
<i>Text formatting</i>	Easy interface and display from 1 to 4 panels	Easy interface for working with text, but doesn't support automatic syn-	To format, you need to know the basic commands	There are 3 modes and they have a different role in creating material	Easy access to text formatting	Easy access to formatting, just master a couple of easy	Just formatting the text is enough	Simple enough, but may take a little time due to the frequent use of the mouse

	Sublime Text	Note-pad++	Vim	Emacs	LyX	LaTeX	MarkHub	MS Word
		tax correction				commands		to text formatting
<i>Insert videos and images</i>	Perhaps, but only with the support of plugins	Supported by plugins	It is possible to add video and photo files	Supported by plugins	Supported by plugins	Yes	Yes (without leaving the book)	It is possible to insert an image, but video does not support
<i>Interactivity</i>	Supported by plugins	Supported by plugins	Maybe depending on language you use	Yes	Yes, you can add video or photo	Yes, they are mainly used to create slides and documents	Yes	Only direct links or images are possible
<i>Document format</i>	Saved locally in the editor	Supports html, css, php, js and many other formats	Supports html, css, php, js and many other formats	Saved to disk or locally. Doesn't interact with GitHub	Saved in different formats	Convert dvi to pdf	pdf, SCORM, epub, universal web book link generation	pdf, docx, xml, html, txt
<i>LaTeX support</i>	Yes	Yes	Yes	Perhaps, with limitations	Yes	No	Yes	Perhaps, with various apps

4 MarkHub Cloud Online Editor

Thus, an analysis of the editors and the state of the market for educational content creation technologies allowed us to offer a cloud-based online editor, MarkHub (<https://markhub.io>), which allows you to generate web books using Markdown markup language with advanced native notation. Investigating other editors, it is clear that modern approaches and technologies take a lot of time when designing a product. This is a major benefit of our online editor, with the MarkHub online editor not only simplifying the interface but also shortening the time when working with MarkHub.

The main advantages of the proposed solution over other editors are:

- extended Markdown notation;
- two windows for work, one of which you write, the other reformat your text in a proper visual appearance;
- the ability to work together;
- sync content with Google Drive and GitHub;
- support for media, audio files, images, tables, direct links, tests;
- export in SCORM, PDF, EPUB formats;

- help material in the form of tips for content collection;
- versatility of browsing the web through your phone, tablet or laptop.

The main element around which MarkHub's cloud-based online editor is built is Markdown Markup Language [19]. Today, the use of this tool affects both the competitiveness of authors and gives wide opportunities to users of such popular systems as: Wikipedia, GitLab, Dropbox, Slack and many others. Turning to the benefits of Markdown, I would like to say that those who first encounter this tool, within a few hours, get used to its functionality and logic [2]. Now, the main thing is, why did we put this system behind the editor:

1. Convenience of distribution

In general, documents written with Markdown are a standard TXT file that is open in all applications. It should be noted that the TXT file will not have a visual design, it will require editors such as MarkHub or StackEdit. The document will only contain commands for the editor. Below you will see an example of how a TXT file was modified with the Markhub editor.

2. Simplicity

The syntax of this technology is very simple. When writing, you need to use characters such as: # (grid), - (dash), * (asterisk), and so on. For example, if you want the author to make a title in a document using MarkHub, just put # - # in front of the title.

3. Rigor and predictability

One of the reasons why we put Markdown technology at the heart of the editor is the existence of strict rules that cannot be ignored – service marks.

To sum up, Markdown technology has strict rules for content design. This can be considered a benefit, because the author who worked with MarkHub notation will easily understand the design of the material. In addition, the result will be predictable and live up to expectations with a clear and simple notation. If small bugs do occur, they can be quickly identified through a preview mode that automatically transforms the commands into visual content visualization. Thus, Markdown is a popular and progressive way to quickly design content that, thanks to its versatility and simplicity, saves considerable time when writing technical text, scientific articles or educational materials. In turn, such advanced editors as MarkHub will significantly simplify and diversify the text based on Markdown principles and principles [12].

As for tools inside the online editor, there are many options for creating content:

- text;
- lists (ordered and unordered);
- definitions and quotations;
- drawings from external sources (photographs, diagrams and diagrams in different notations, mathematical graphs and diagrams);
- circuits built using standard blocks and connectors;
- tables;

- mathematical formulas;
- literal blocks (fragments of printouts, program listings, etc.);
- links and footnotes;
- media content;
- tests.

Let's take a look at the steps of preparing educational content in an e-book format in the cloud-based online editor MarkHub. The user interface in the editor is built using the WYSIWYM concept – the main elements of the interface are the source editor (left in the center) and the preview area (right in the center) (Fig. 1). The user in the editor works with the source text (content), applying to it the rules of Markdown markup language, and in the preview area, continuous rendering (formation “on the fly”) of the appearance of the content is carried out.

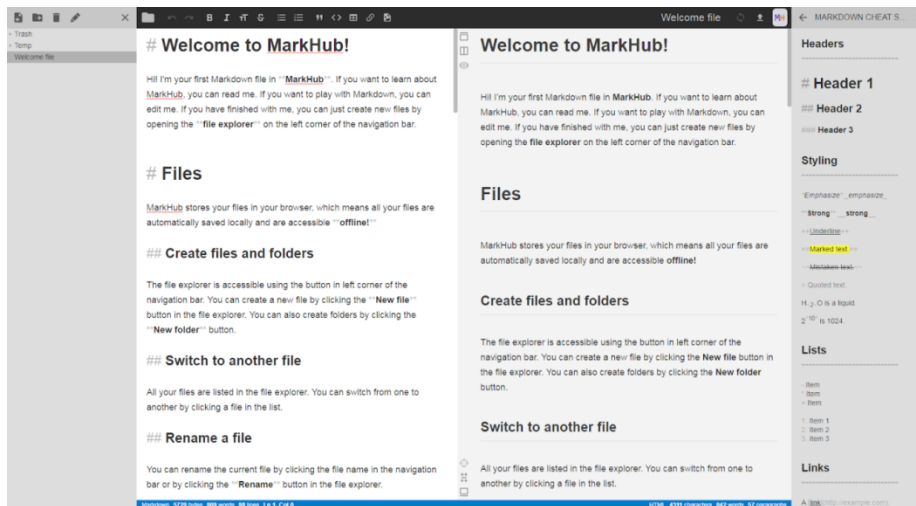


Fig. 1. MarkHub Cloud Online Editor User Interface

At any point in time, the user can use the Build button on the top toolbar on the right, which allows them to generate an e-book and place it online on MarkHub server resources for further dissemination and use through the link. The appearance of the generated web book is shown in Fig. 2.

The proposed toolkit in a single MarkHub platform provides a huge impetus for competitiveness among other authors. Interactivity can easily capture the reader's attention, and the easy content creation functionality will reduce the time spent on routine operations by 30% and move on to the more significant tasks of the learning process.

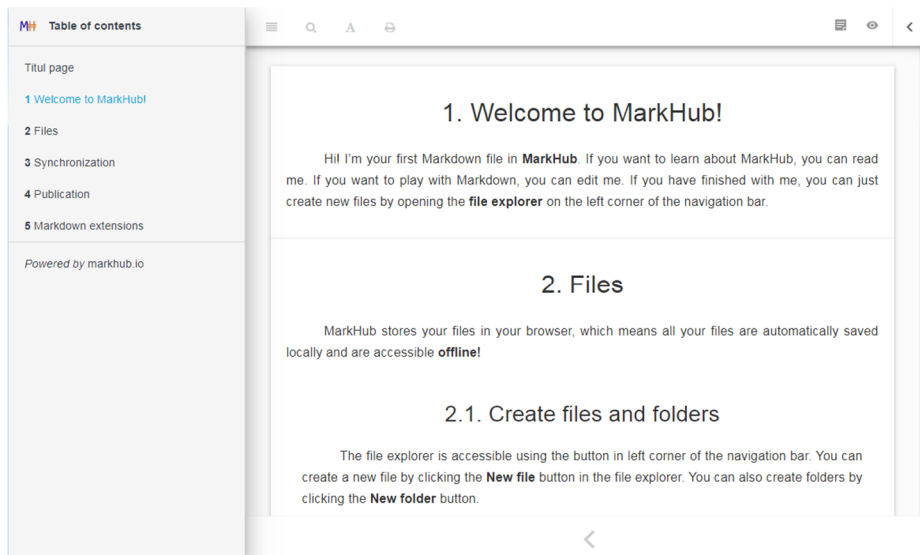


Fig. 2. An example of a finished web book after creating content in the editor

5 Conclusions

Using the WYSIWYM approach to unifying educational content development can significantly increase the author's productivity in the content creation process. This becomes possible on the basis of:

- reducing the cost of time to format the external presentation of content,
- making synchronous changes to different versions of content,
- tracking versions of content,
- organization of remote teamwork in a network environment.

All these workflows have a high proportion in the author's educational activities, so their optimization is important in the current conditions of rapid creation and dissemination of information in the world.

References

1. Babenko, V., Lomovskykh, L., Oriekhova, A., Korchynska, L., Krutko, M., Koniaieva, Y.: Features of methods and models in risk management of IT projects. *Periodicals of Engineering and Natural Sciences* **7**(2), 629–636 (2019). doi:10.21533/pen.v7i2.558
2. Babenko, V.: Formalization of the Model of Management of the Technological Innovations. *CEUR Workshop Proceedings* **2393**, 595–602 (2019)
3. Bond, W., Skinner, J.: Sublime Text 3.1 Released. <https://www.sublimetext.com/blog/articles/sublime-text-3-point-1> (2018). Accessed 25 Oct 2019

4. Carlson, H., Falk D.: Effective Use of Interactive Videodisc Instruction in Understanding and Implementing Cooperative Group Learning with Elementary Pupils in Social Studies and Social Education. *Theory & Research in Social Education* **17**(3), 241–258 (1989). doi:10.1080/00933104.1989.10505591
5. Children of the tech revolution. *The Sydney Morning Herald*. <https://www.smh.com.au/lifestyle/children-of-the-tech-revolution-20080715-gdsma9.html> (2008). Accessed 25 Oct 2019
6. Cocker, A.: A introduction to creating documents in LaTeX. Learn to typeset documents in the LaTeX text markup language. <https://opensource.com/article/17/6/introduction-latex> (2017). Accessed 25 Oct 2019
7. Kay, A., Goldberg, A.: Personal Dynamic Media. *Computer* **10**(3), 31–41 (1977). doi:10.1109/C-M.1977.217672
8. Khalili, A., Auer, S.: WYSIWYM Authoring of Structured Content Based on Schema.org. In: Lin, X., Manolopoulos, Y., Srivastava, D., Huang, G. (eds.) *Web Information Systems Engineering – WISE 2013*. WISE 2013. Lecture Notes in Computer Science, vol. 8181, pp. 425–438. Springer, Berlin, Heidelberg (2013). doi:10.1007/978-3-642-41154-0_32
9. Knuth, D.: Curriculum Vitæ. <https://www-cs-faculty.stanford.edu/~knuth/vita.html>. Accessed 25 Oct 2019
10. Korneev, A.N., Kotelnikova, V.E.: Tendencii v oblasti distancionnog oobucheniia: metody i tekhnologii (Trends in Distance Learning: Methods and Technologies). *Naukovedenie* **9**(6). <https://naukovedenie.ru/PDF/30EVN617.pdf> (2017). Accessed 25 Oct 2019
11. Lamport, L.: Document Production: Visual or Logical? *Notices of the American Mathematical Society* **34**, 621–624 (1987)
12. Leonard, S.: Guidance on Markdown: Design Philosophies, Stability Strategies, and Select Registrations. <https://tools.ietf.org/html/rfc7764> (2016). Accessed 21 Mar 2017
13. LyX – The Document Processor. <https://www.lyx.org> (2020). Accessed 21 Mar 2020
14. Myers, B.A.: A Brief History of Human-Computer Interaction Technology. *Interactions* **5**(2), 44–54 (1998)
15. Ponomarenko, V.S., Klebanova T.S., Yatsenko R.N.: Adaptivnaya sistema distancionnogo obucheniya (The adaptive distance learning system). *BUSINESS-INFORM.* **4**(2), 174–178 (1995)
16. Schäferhoff, N.: Notepad++ Review – A Powerful, Free Code Editor Packed With Features. <https://www.elegantthemes.com/blog/resources/notepad-review-a-powerful-free-code-editor-packed-with-features> (2017). Accessed 25 Oct 2019
17. Son, B.: Getting started with Vim: The basics. <https://opensource.com/article/19/3/getting-started-vim> (2019). Accessed 25 Oct 2019
18. Stella, J.: A CEO’s Guide to Emacs. <https://www.fugue.co/blog/2015-11-11-guide-to-emacs.html> (2015). Accessed 21 Mar 2017
19. Xie, Y., Allaire, J.J., Grolemond, G.: *R Markdown: The Definitive Guide*. Chapman & Hall, Boca Raton (2018)