|  |
| --- |
|  |
|  |
| Microsoft Word  Introduction to Configuration |
| **Maciej Słojewski** |

Change log

| Version | Date | Change |
| --- | --- | --- |
| 1 | 2025-08-24 | The first issue of this document. |

Published without authorization of Microsoft Corporation.

Published by Maciej Słojewski. For contact with author search the [LinkedIn](https://www.linkedin.com/in/maciejslojewski/) profile of the author.

Copyright © 2025 by Maciej Słojewski

Copyright: not all rights are reserved. [CC BY-NC-SA](https://en.wikipedia.org/wiki/Creative_Commons_license) = Creative Commons (CC) Attribution (BY) Non-commercial (NC) SA (Share-alike).

No patent liability is assumed with respect to the use of the information contained herein. Although every precaution has been taken in the preparation of this book, the publisher and author assume no responsibility for errors or omissions. Nor is any liability assumed resulting from the use of the information contained herein.

ISBN: (…)

Trademarks

Microsoft and the trademarks listed at <http://www.microsoft.com> on the “Trademarks” webpage are trademarks of the Microsoft group of companies. All other marks are the property of their respective owners.

Warning and Disclaimer

Every effort has been made to make this book as complete and as accurate as possible, but no warranty or fitness is implied. The information provided is on an “as is” basis. The author, the publisher shall have neither liability nor responsibility to any person or entity with respect to any loss or damage arising from the information contained in this book or from the use of the programs accompanying it.

GitHub and source code license

The source code of macros discussed in this book is available at   
<https://github.com/mslonik/Microsoft-Word-Configuration>

The source code license is MIT:

Copyright (c) 2025 Maciej Słojewski

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

List of Content

1. [Change log 1](#_Toc219486239)
2. [List of Content 3](#_Toc219486240)
3. [1. Introduction, why did I write this book? 6](#_Toc219486241)
   1. [1.1. Less is more 6](#_Toc219486242)
   2. [1.2. How to read this book 6](#_Toc219486243)
   3. [1.3. The main drawbacks of Microsoft Word 6](#_Toc219486244)
   4. [1.4. You've been warned… 7](#_Toc219486245)
   5. [1.5. Tested on 8](#_Toc219486246)
   6. [1.6.  Disclaimer 8](#_Toc219486247)
4. [2. What is Microsoft Word configuration? 9](#_Toc219486248)
   1. [2.1. Examples 9](#_Toc219486249)
      1. [2.1.1.  Configuration files 9](#_Toc219486250)
      2. [2.1.2. Application configuration 9](#_Toc219486251)
5. [3. Strategies for configuration file management 10](#_Toc219486252)
   1. [3.1. So where do I store styles and style definitions? 10](#_Toc219486253)
6. [4. Configuration folders and files 11](#_Toc219486254)
   1. [4.1. User customizable configuration files 13](#_Toc219486255)
      1. [4.1.1. Location of template files (A and B) 14](#_Toc219486256)
      2. [4.1.2. Location of theme file 16](#_Toc219486257)
      3. [4.1.3. Menu file 17](#_Toc219486258)
   2. [4.2. External configuration files and folders 17](#_Toc219486259)
   3. [4.3. Environment, backup operation files and folders 19](#_Toc219486260)
   4. [4.4. Environment, Microsoft Word and OneDrive 19](#_Toc219486261)
7. [5. What are the template files? 23](#_Toc219486262)
   1. [5.1. What kind of information may be stored within a template file? 23](#_Toc219486263)
   2. [5.2. Attachment relationship 24](#_Toc219486264)
   3. [5.3. What are the other relationships beside the attachment? 26](#_Toc219486265)
   4. [5.4. What does configuration "context" mean in Microsoft Word? 27](#_Toc219486266)
      1. [5.4.1. When does a context matter? 27](#_Toc219486267)
   5. [5.5. How do you make the *attachment* relationship between a content file and template manually? 28](#_Toc219486268)
   6. [5.6. How the tick *Automatically update document styles* works? 28](#_Toc219486269)
   7. [5.7. Can I use a template file as stencil or content template? 30](#_Toc219486270)
8. [6. What are the theme files? 31](#_Toc219486271)
   1. [6.1. What are the menu files? 31](#_Toc219486272)
   2. [1.1. 31](#_Toc219486273)
9. [7. Content files 32](#_Toc219486274)
   1. [7.1. Recommended information categories 33](#_Toc219486275)
   2. [7.2. What happens to formatting if I move my content file to another computer? 33](#_Toc219486276)
10. [8. Configuration of Microsoft Word Graphical User Interface 34](#_Toc219486277)
    1. [8.1. Design of Microsoft Word GUI 34](#_Toc219486278)
    2. [8.2. Ribbon menu 35](#_Toc219486279)
    3. [8.3. Status bar 35](#_Toc219486280)
    4. [8.4.  Panes 37](#_Toc219486281)
11. [9. Running Microsoft Word 38](#_Toc219486282)
    1. [9.1. Tryb pracy edycja dokumentu, scenariusze 38](#_Toc219486283)
       1. [9.1.1. Uruchomienie samej aplikacji Microsoft Word 40](#_Toc219486284)
       2. [9.1.2. Uruchomienie Microsoft Word poprzez wybór istniejącego pliku 41](#_Toc219486285)
       3. [9.1.3. Uruchomienie Microsoft Word poprzez wybór pliku szablonu 42](#_Toc219486286)
    2. [9.2. Tryb pracy edycja plików konfiguracyjnych 42](#_Toc219486287)
    3. [9.3. Normal.dotm 43](#_Toc219486288)
12. [10. Microsoft Word processor workflows 44](#_Toc219486289)
13. [11. Informacje składowe wykorzystywane przez Microsoft Word 45](#_Toc219486290)
    1. [11.1. Content 45](#_Toc219486291)
    2. [11.2. Style (C\_F\_Styles.dotm) 45](#_Toc219486292)
       1. [11.2.1. Why to make this file? 45](#_Toc219486293)
       2. [11.2.2. How to make this file? 46](#_Toc219486294)
       3. [11.2.3. Jak wygląda klasyfikacja typów styli? 46](#_Toc219486295)
       4. [11.2.4. Quirks of default styling "Set Defaults" 46](#_Toc219486296)
       5. [11.2.5. Paragraph settings, line spacing 48](#_Toc219486297)
       6. [11.2.6. Jakie są najważniejsze skróty klawiaturowe związane z wykorzystaniem styli? 48](#_Toc219486298)
       7. [1.1.1. Jak usunąć zbędne style z szablonu? 49](#_Toc219486299)
       8. [11.2.7. Czy do pliku DOCX da się dołączyć szablon bez automatycznego kopiowania styli? 49](#_Toc219486300)
       9. [11.2.8. Dlaczego ktoś może chcieć to zrobić? 50](#_Toc219486301)
       10. [11.2.9. Jak sprawdzić, czy w dokumencie prawidłowo wykorzystane są style z szablonu? 50](#_Toc219486302)
       11. [11.2.10. Czy gdzieś można zobaczyć, jak wyglądają wszystkie style znajdujące się w szablonie? 50](#_Toc219486303)
       12. [11.2.11. Projekt stylu „Normalny ms” 50](#_Toc219486304)
       13. [11.2.12. Style nagłówków 51](#_Toc219486305)
       14. [11.2.13. Jak projektować style? 51](#_Toc219486306)
    3. [11.3. Makra (Macros\_ms.dotm) 51](#_Toc219486307)
       1. [11.3.1. Jak sprawdzić, czy kod makr daje się skompilować 52](#_Toc219486308)
       2. [11.3.2. 52](#_Toc219486309)
       3. [11.3.3. Printing options 52](#_Toc219486310)
    4. [11.4. Fonts 53](#_Toc219486311)
       1. [11.4.1. Where fonts file are located? 53](#_Toc219486312)
       2. [11.4.2. How do you check if some fonts aren’t missing in my document? 53](#_Toc219486313)
       3. [11.4.3. How to embed fonts within your document? 54](#_Toc219486314)
       4. [11.4.4. Default font 54](#_Toc219486315)
       5. [11.4.5. Features of *Excalifont-Regular.ttf* fonts 54](#_Toc219486316)
       6. [11.4.6.  Font math 54](#_Toc219486317)
       7. [11.4.7. Skąd pobrać czcionki *Excalifont-Regular.ttf*? 55](#_Toc219486318)
       8. [11.4.8. Jak wygląda rodzina czcionek *Excalifont-Regular.ttf*? 55](#_Toc219486319)
       9. [11.4.9. Czy proces osadzania czcionek jest odwracalny? 55](#_Toc219486320)
       10. [11.4.10. Fonts management 55](#_Toc219486321)
    5. [11.5. Theme (Theme\_ms.thmx) 56](#_Toc219486322)
    6. [11.6. Keyboard shortcuts 58](#_Toc219486323)
       1. [11.6.1. What are the keyboard shortcuts? 58](#_Toc219486324)
       2. [11.6.2. Why keyboard shortcuts in Microsoft Word? 58](#_Toc219486325)
       3. [11.6.3. Is there an alternative approach? 59](#_Toc219486326)
       4. [11.6.4. What are advantages and disadvantages of keyboard shortcuts? 59](#_Toc219486327)
       5. [11.6.5. What have I done to make your life easier? 59](#_Toc219486328)
       6. [11.6.6. Keyboard shortcuts chaining 61](#_Toc219486329)
       7. [11.6.7. Keyboard shortcuts classification in Microsoft Word 61](#_Toc219486330)
       8. [1.1.2. Keyboard shortcuts and context 62](#_Toc219486331)
       9. [11.6.8. Listing of all customized shortcuts and relationship to macros 62](#_Toc219486332)
       10. [11.6.9.  Copying of all customized shortcuts to TXT files 63](#_Toc219486333)
       11. [11.6.10.  List of all commands and related keyboard shortcuts 63](#_Toc219486334)
14. [12. Jak wygląda relacja szablon DOTM ↔ plik DOCX 64](#_Toc219486335)
    1. [12.1. Co się dzieje po dołączeniu do pliku DOCX pliku szablonu (DOTM)? 64](#_Toc219486336)
    2. [12.2. Co dzieje się po odłączeniu lub niedostępności szablonu? 64](#_Toc219486337)
    3. [12.3. Prawda czy fałsz: Definicje stylów są przechowywane w pliku DOCX 64](#_Toc219486338)
    4. [12.4. Czy style zawsze są poprawnie przenoszone z szablonu do pliku? 64](#_Toc219486339)
    5. [12.5. Co zrobić, jeżeli definicja stylu nie zostaje przeniesiona do pliku z szablonu? 65](#_Toc219486340)
15. [13. Pliki i prawo 66](#_Toc219486341)
    1. [13.1. Pliki dokumentów (DOCX) i prawo 66](#_Toc219486342)
       1. [13.1.1. W jaki sposób chronimy treść w dokumentach na bazie szablonów? 66](#_Toc219486343)
       2. [13.1.2. Jaką klasę poufności wybrać? 66](#_Toc219486344)
       3. [13.1.3. Nadal nie wiem, jaką klasę poufności wybrać, co mam zrobić? 66](#_Toc219486345)
       4. [13.1.4. W jakiej formie wysłać dokument do klienta? 67](#_Toc219486346)
       5. [13.1.5. W jaki sposób wyjaśnić klientowi oznaczenie klasy poufności? 67](#_Toc219486347)
       6. [13.1.6. Nie jestem pewna / pewny, czy mogę wysłać dany dokument. Co mam zrobić? 67](#_Toc219486348)
    2. [13.2. Pliki szablonów (DOTM) i prawo 68](#_Toc219486349)
       1. [13.2.1. Czy szablony są objęte prawem autorskim? 68](#_Toc219486350)
       2. [13.2.2. Czy wysyłając do klienta plik szablonu wysyłam mu jednocześnie czcionki? 68](#_Toc219486351)
16. [2.  Design of configuration 69](#_Toc219486352)
17. [14. Dlaczego pliki szablonów często się zmieniają? 70](#_Toc219486353)
    1. [14.1. W jaki sposób zapewnić aktualizację wszystkich składników szablonu w konkretntym dokumencie? 70](#_Toc219486354)
    2. [14.2. Kto jest odpowiedzialny za utrzymanie plików szablońow? 70](#_Toc219486355)
    3. [14.3. Czy pliki szablonów są nadzorowane przez system kontroli wersji? 70](#_Toc219486356)
18. [15. Ustawienia strony 71](#_Toc219486357)
    1. [15.1. Marginesy 71](#_Toc219486358)
    2. [15.2. Wielkości marginesów 71](#_Toc219486359)
    3. [15.3. Podział na wyrazy 71](#_Toc219486360)
19. [16. Listy 72](#_Toc219486361)
    1. [16.1. Pułapki 72](#_Toc219486362)
       1. [16.1.1. Jak sobie z tymi pułapkami poradzić? 72](#_Toc219486363)
       2. [16.1.2. Wyłączenie automatycznej numeracji list 72](#_Toc219486364)
    2. [16.2. Jak ręcznie nadawać style dla list? 76](#_Toc219486365)
       1. [16.2.1. Krok 0: akapity tekstu z domyślnym stylem akapitu 76](#_Toc219486366)
       2. [16.2.2. Krok 1: nadanie akapitom listy właściwego stylu „Połączone (akapit i znak)” 77](#_Toc219486367)
       3. [16.2.3. Krok 2: nadanie akapitom listy właściwego stylu „Lista” 77](#_Toc219486368)
    3. [16.3. Jak powinna wyglądać konfiguracja stylu „Wypunktowanie ms”? 78](#_Toc219486369)
       1. [16.3.1. Pułapki 79](#_Toc219486370)
    4. [16.4. Jak powiązać pola z wypunktowaniami? 82](#_Toc219486371)
       1. [16.4.1. Definiujemy pole SEQ 82](#_Toc219486372)
       2. [16.4.2. Definiujemy blok konstrukcyjny 82](#_Toc219486373)
    5. [16.5. Zasady stosowane dla list numerowanych i nienumerowanych w szablonach 83](#_Toc219486374)
20. [17. Tabele 84](#_Toc219486375)
    1. [17.1. Jak formatować tabele? 86](#_Toc219486376)
    2. [17.2. Style typu „tabela” a szablon 87](#_Toc219486377)
21. [18. SVN 88](#_Toc219486378)
    1. [*18.1.* Kod skryptu *bash* po stronie serwera *post-commit.sh* 88](#_Toc219486379)
    2. [18.2. Kod skryptu *bash* po stronie serwera *pre-commit.sh* 89](#_Toc219486380)
    3. [18.3. Kod makra VBA *InsertSVNCommitNumber* 89](#_Toc219486381)
22. [19. This book repository 92](#_Toc219486382)
    1. [19.1. Structure 92](#_Toc219486383)
23. [20. Backup of working files 93](#_Toc219486384)
24. [21. Jak szukać pomocy do Microsoft Word? 94](#_Toc219486385)
    1. [21.1. Oficjalna pomoc online 94](#_Toc219486386)
    2. [21.2. Oficjalne forum 94](#_Toc219486387)
    3. [21.3. Zgłaszanie nieprawidłowości w Microsoft Word 94](#_Toc219486388)
    4. [21.4. Official releases and known bugs 94](#_Toc219486389)
25. Introduction, why did I write this book?

Other questions to be addressed: Why did I write another book about Microsoft Word? What is so special about it to create next book?

I have years of practice in a corporate environment, where Microsoft Word was the only one tool available for authoring all kinds of documentation: technical documentation, formal documentation, user documentation. No matter how you would put division line between different kinds of documentation, only Microsoft Word was allowed to create files.

Microsoft Word is a so called "word processor", to distinguish it from ordinary text editors and underline it is a feature reach, more like desktop publishing program. Additionally, it is WYSIWYG tool. This long acronym stands for What You See Is What You Get. At the time unique feature which enabled printed out content look the same as on the screen on time of authoring.

Be warned: there are alternatives for authoring documentation. Perhaps more straightforward, with a smaller number of exceptions, more intuitive, more convenient to use. I did not have a choice. in consequence I paid hundreds of hours to customize Microsoft Word to follow my requirements. And this book is about this customization.

To be fair, every tool you would like to use professionally for content authoring would require plenty of additional work dedicated to customization. I still wonder, if hundreds, or likely thousands. I couldn't find one complete introduction or manual at least main intricacies of Microsoft Word configuration. So, I decided to write one on my own. Of course, it is true that I did it for me. But why not share it, if it is ready?

There is more general question, why write a book about configuration of a specific application? Firstly, the content of a book would depend on changes to this application. In other words, it can rapidly become out of date after the next update. Secondly, it is quite subjective, as configuration would be dedicated to specific needs or applications. From one perspective all software is just a configuration to hardware. Let's keep this idea in mind.

Why a configuration matters? Because at least in case of Microsoft Word in can significantly reduce time necessary in any way to make content look right. The previous sentence doesn't sound too substantial. For example, what does "significantly" mean? I mean multiple times faster: two times at least. Thanks to limited set of styles, and keyboard shortcuts assigned to styles, Microsoft Word can really shine. Of course, it requires some upfront work (investment) related to configuring a set of styles and keyboard shortcuts. It pays off, and you can even try to estimate it quantitatively, if you wish. And that example is just one of many you can find in this book.

The second point which I'd like to share with you is some simple advice regarding the design of documents. Even more subjective than configuration, based on my personal experience.

* 1. Less is more

Microsoft Word offers a tremendous number of functions, options, choices. It is like Swiss pocketknife equipped in more than hundreds of simple blades or tools. And when you finally have found one necessary for the accomplishment of your job, you realize that to make that job convenient, you would like to separate it from the rest of your knife-set, but it is not possible. As a result, you have found the right tool, but there is no easy way to make use of it.

I propose in this book another approach: less is more. I try to cut off as much as possible all not relevant functions, elements of interfaces, styles or whatever, to leave you with something simple to do your job.

* 1. How to read this book

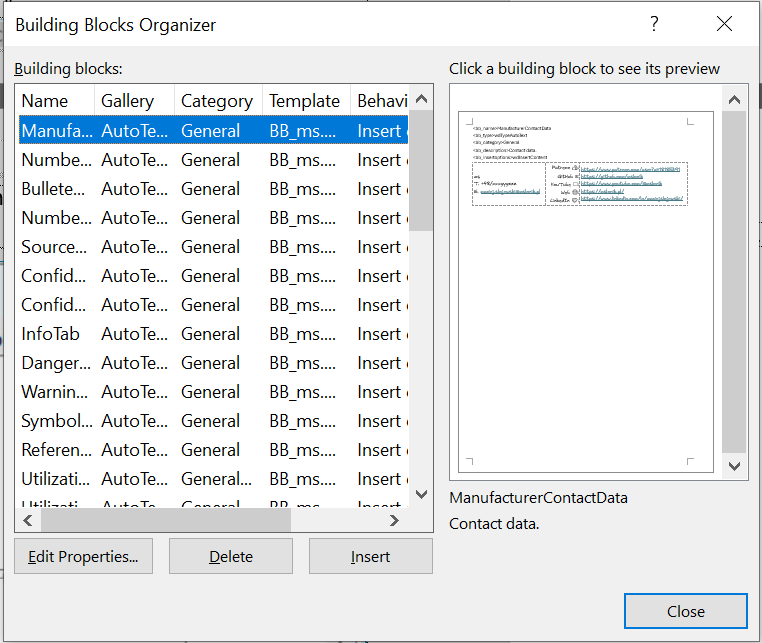
(…)

* 1. The main drawbacks of Microsoft Word

My main point: I was promised a possibility to manage this application from Visual Basic for Applications (VBA). But this promise is covered only partially. On the one hand it is a wonderful tool, on the other hand it is a curse, as code cannot be easily managed.

Below you can find my subjective list of main drawbacks of Microsoft Word:

* + Many frequently used dialog windows are small, and you can't change its size. There is no window scaling (see Pic. 1).
  + Many areas of Microsoft Word are not accessible from VBA code. Example: status bar.
  + For many options there is no "default" setting which should enable us to restore initial configuration.
  + Shortcut management is cumbersome, to say at least.
  + Autotext definitions (text strings) cannot be applied to call a macro.
  + It is impossible to move some settings of the content, e.g. sizes of margins, distances to footer and header between different documents, without applying the VBA macros.



Pic. 1. Example of small dialog box. [source: herein]

* 1. You've been warned…

If you don't need to use Microsoft Word, don't use it. There are alternatives, but… other applications would also require from you a lot of effort to prepare valid and useful configuration. Hopefully with less frustration.

And now, if you aren't yet discouraged or offended, or basically just curious what is in the content of this book, I sincerely invite you to read on.

* 1. Tested on

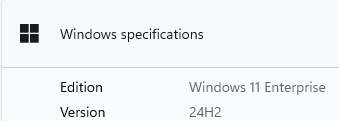
All the macros, as well as template files, have been tested on two PCs with slightly different configuration, as specified in Tab. 1 and Tab. 3.

Tab. 1. First tested setup. [source: herein]

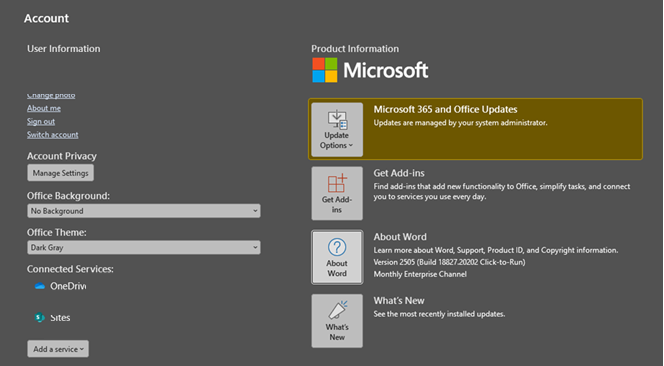
|  |  |  |
| --- | --- | --- |
| Feature | Microsoft Windows | Microsoft Word |
| Edition | Windows 10 Home | Microsoft® Word 2016 MSO 64-bit |
| Version | 22H2 | Version 2507 |
| OS Build | 19045.6216 | Build 16.0.19029.20136 |

Tab. 2. Second tested setup. [source: herein]

|  |  |  |
| --- | --- | --- |
| Feature | Microsoft Windows | Microsoft Word |
| Edition | Windows 11 Enterprise | Microsoft® Word for Microsoft 365 MSO 64-bit |
| Version | 24H2 | 2505 |
| OS Build | 26100.4652 | 16.0.18827.20102 |



Pic. 2. Microsoft Windows: Control Panel > System > About > Windows specifications.. [source: herein]



Pic. 3. Microsoft Word: File > Account > About Word. [source: herein]

* 1. Disclaimer

This book is only an introduction to Microsoft Word configuration. It is also work in progress. I’m pretty sure you’ll be able to improve it and adjust to your own needs.

No liabilities. Use "as is", including VBA source code.

1. What is Microsoft Word configuration?

Configuration is a set of parameters to set up state of Microsoft Word, such as:

1. Configuration files,   
   files which store customized configuration and can be safely moved in form of files. These files contain information important to formatting content and operation of Microsoft Word.
2. Microsoft Word application configuration,   
   stored within Microsoft Windows registry, can be saved in the form of files, but moving it to another computer requires additional precaution.

Unfortunately, Microsoft Word configuration is not ready to be easily moved from one PC to another. I did as much as possible to make it possible by applying VBA macros stored in configuration files. Nevertheless, there are still strange quirks which in a specific situation require manual settings, as not all GUI options are available from VBA code.

* 1. Examples
     1. Configuration files

The Normal.dotm file, named a template file, stores the default set of styles. The styles influence setup of look and feel of the content. No matter if you know about it, or consciously make a choice, any content present on a Microsoft Word screen, some styles are applied. By default, styles come from Normal.dotm file. It means, that by default, if no other template file is linked to a content file, that one is applied. The Normal.dotm file by default is stored within a folder:

C:\Users\<YourUserName>\AppData\Roaming\Microsoft\Templates\

It can be copied to another PC, to another user profile.

* + 1. Application configuration

Generic configuration of Microsoft Word, I mean independent of content. For example, how is content displayed on a screen: File > Options > Advanced > Image Size and Quality. The Default resolution parameter cannot be set from VBA. Probably it is stored within Microsoft Windows registry:

HKEY\_CURRENT\_USER\Software\Microsoft\Office\16.0\Word

But where exactly, I cannot say as it is not documented.



Pic. 4. Example. [source: herein]

1. Strategies for configuration file management

The idea that configuration is stored independently from content is usual practice in modern computing but still may require some comments.

In general, there are two competing strategies. Thanks to flexibility of Microsoft Word, both can be applied at the same time:

1. modular:  
   single file with the content +  
   one or more files containing configuration
2. minimal:   
   single file with content and configuration.

Tab. 3. Comparison of strategies for content configuration. [source: herein]

|  |  |  |  |
| --- | --- | --- | --- |
| strategy | advantages | disadvantages | comments |
| modular | Separation of different types of information into files.  Modularity, flexibility.  Crucial files are relatively smaller.  Dedicated maintenance tasks to team members. | Maintenance of many files | Author's choice. |
| minimal | Easy to copy / send out. | No logical separation. If something is wrong, the whole file is affected. | It could be relatively easily made from a modular model. |

This book describes the modular strategy, only briefly touching the other one. Practical approach:

* + Most of my macros I keep in a separate file, which is always, by default, loaded by Microsoft Word on startup Macros\_ms.dotm. Thanks to that no matter what content file I work on now, I can apply the same set of macros or just switch them off.
  + I keep theme settings in a separate file Theme\_ms.thmx. Thanks to that I can apply e.g. specific colors to any file containing styling.
  + I keep building blocks in a separate file BB\_ms.dotm. Thanks to that I can use them as a module, for a dedicated purpose or just switch them off.
  + I keep User Interface (UI) settings in a separate file RibbonCustomizations\_ms.exportedUI. There are stored menu buttons which enable quick access to majority of macros in Macros\_ms.dotm.
  1. So where do I store styles and style definitions?

I store them within my macros file, Macros\_ms.dotm. There are numerous advantages of this approach:

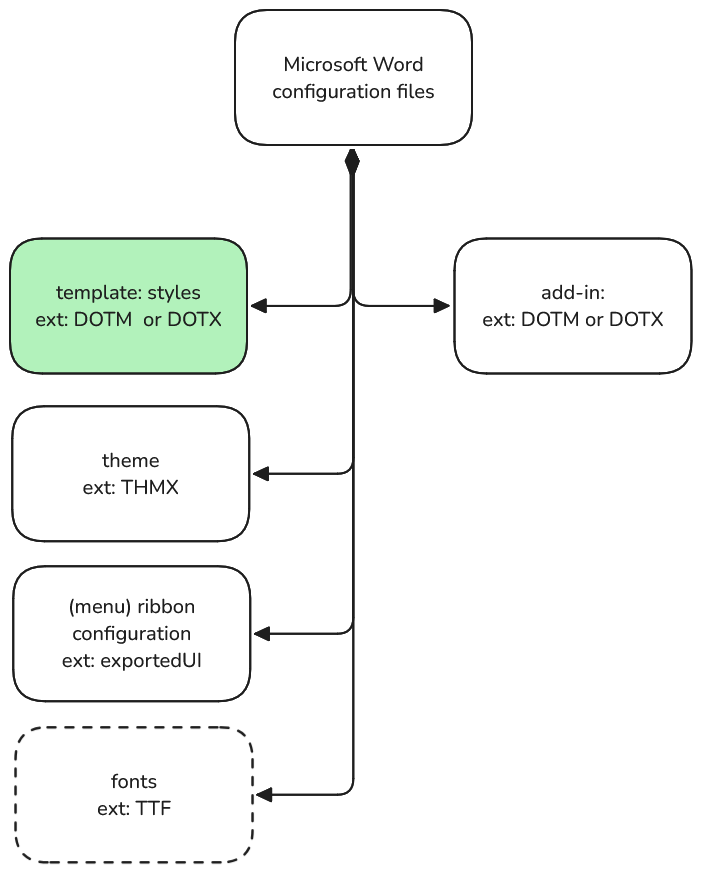
* Styles are always stored within content file (e.g. DOCX) and must be copied there, no matter if consciously or not. Thanks to storage in macros plug-in I have to consciously move them to my content file.
* Text definition of all styles makes it easy to trace changes, apply them uniformly to specific styles. It is fast and elegant.
* To move my styles from one file to another, all that is required is to… run specific macro, which copies all customized styles.

Following principle “less is more” I don’t use any built-in styles of Microsoft Word if I don’t have to. There is one important exception to this rule: built-in styles Table of Content.

1. Configuration folders and files

In this chapter we will meet configuration files, see in Pic. 5. Whether we like it or not, the conclusion is there are several types and format of the file which can be involved in the process of configuring the Microsoft Word. These configuration files create an environment for Microsoft Word operation.

The first criteria used here are the type of a file, understood as file format and file extension. In each category there can be numerous numbers of files. In the following subchapters I will show first where the files are stored and next describe what special features each category has. If you don't want to dive too deep, all you need to start is location of configuration files in chapter 4.1 and then you can skip the rest of this chapter.



Pic. 5. Microsoft Word configuration files. [source: MicrosoftWordTemplateConcept.excalidraw]

Legend:

* + The green color represents the only one mandatory category, the templates.
  + The dashed line represents files which are technically not configuration files, but may be helpful to get expected look and feel. The only practical case will be to integrate font files within DOCX file if it is edited on PC where specific fonts are not available. Then Microsoft Word enables integration of font files with DOCX file.
  + The ext stands for file extension.

The file with extension DOTM or DOTX can be named as template or add-in. The distinguishing between these two is essential and “designed” by creators of Microsoft Word, but not obvious and somehow hidden. To some degree you can recognize “which is which” by its location.

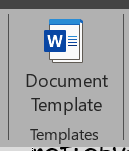
When Microsoft Word is started by clicking icon “Word” or making deliberate choice from within Start Menu, and application configuration wasn’t altered in any strange way, it opens “Document1”, what is visible in the middle of Microsoft Word main window top bar, see Pic. 9. In the meantime Microsoft Word already “attached” the default template file Normal.dotm and copied all built-in styles (> 500 items) to this file.



Pic. 6. Example. [source: ]

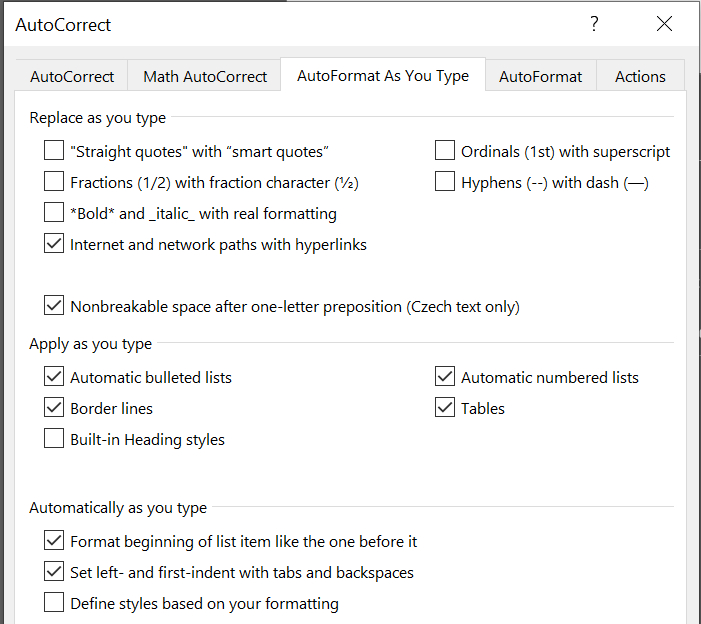
There is alternative way of running Microsoft Word, preferred by many users. They locate default Microsoft Word template file authored by company, e.g. MyCompanyTemplateFile.dotm and run it as if it was an executive file (EXE). Then Microsoft Word starts and again “Document1” is opened, but this time list of styles contains: all built-in styles (> 500 items) plus specific styles defined in this MyCompanyTemplateFile.dotm. But user is not informed about it.

The only way to at least have a feeling what is going on in User Interface with attached template file is to investigate ribbon menu Developer > Templates > Document Template button (see Pic. 10). By default (but you can change if, if you wish) it is located in the hidden menu Developer.



Pic. 7. Document Template button. [source: herein]

| **🄸** Info |
| --- |
| To make ribbon menu Developer visible: |



Pic. 8. Example. [source: ]

* 1. User customizable configuration files

My goal is to make use of built-in automatic mechanisms of Microsoft Word to manage my customized configuration files.

Let me redraw the Pic. 5 with greater detail, adding to each category specific files, resulting in Pic. 9. I’ve also numbered all the files except font files. As you can see there are in total up to 7 files organized in up to 5 folders (A to E).

Microsoft Word  
configuration files

document template files, folder A

Normal.dotm (1)  
(customized macros)

Styles\_ms.dotm(2)  
(customized styles, customized macros)

MicrosoftDefaultBB.dotx (3)  
(default Microsoft BuildingBlocks)

global templates and add-ins, folder B

Macros\_ms.dotm(5)  
(customized macros)

Macros\_ms.dotm(6)  
(customized building blocks)

theme file(s), **folder C**

Macros\_ms.dotm(4)

menu (ribbon)   
configuration file(s),   
**folder D**

RibbonCustomizations\_ms.exportedUI(7)  
customized menu

font files, **folder E**

Excalifont-Regular.ttf

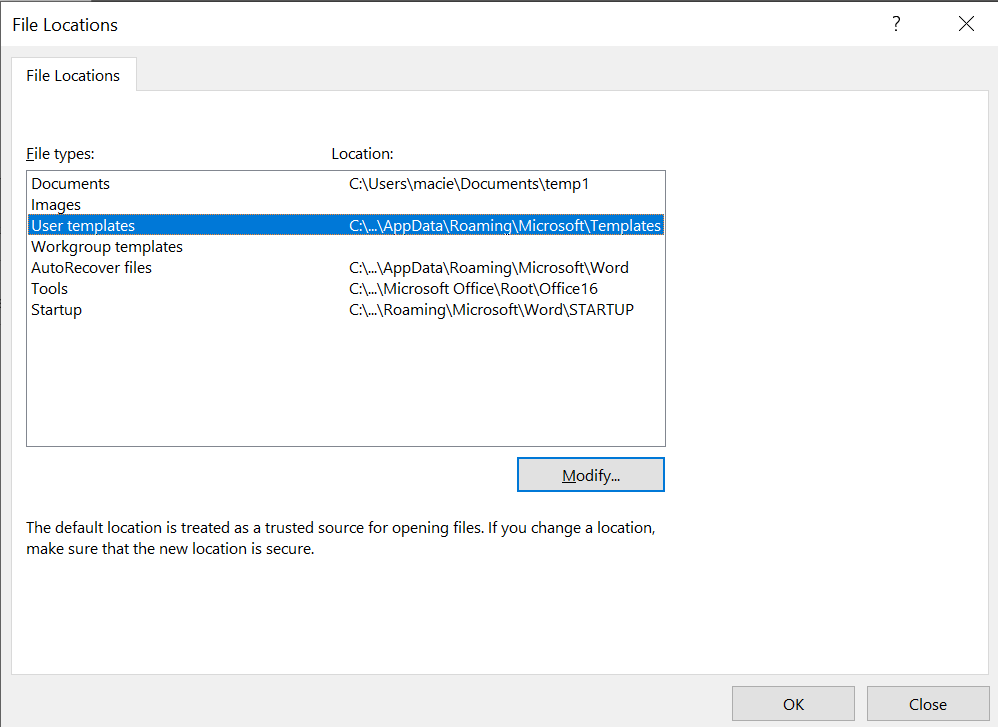
Pic. 9. Proposed organization of configuration files. [source: herein]

Tab. 4. Organization of configuration files. [source: herein]

| folder name and location | file | content |
| --- | --- | --- |
| template folder A | Normal.dotm (1) | Customized macros. |
| (2) | It wasn't possible to store within this file customized styles only. There are also customized macros. |
| MicrosoftDefaultBB.dotx (3) | These are building blocks delivered within Microsoft Word by Microsoft. See chapter (…) for further details |
| template folder B | (5) | Main set of customized macros. |
| (6) | Customized building blocks. |
| folder C:  C:\Users\v523580\AppData\Roaming\Microsoft\Templates\Document Themes\ | Theme\_ms.thmx  (4) | Customized theme (fonts, colors). |
| folder D:  (…) | RibbonCustomizations\_ms.exportedUI (7) | Customized menu (ribbon). |
| folder E:  C:\Windows\Fonts.  C:\Users\<UserName>\AppData\Local\Microsoft\Windows\Fonts | Excalifont-Regular.ttf | C:\Windows\Fonts directory is the primary location for fonts that are available to all users on the computer. However, in more recent versions of Windows (Windows 10 and 11), a second location has become common for user-specific font installations.  So, while C:\Windows\Fonts remains the classic and most important location for system fonts, it's not the only place where font files may be stored. |

* + 1. Location of template files (A and B)

For your disposal you have available the following set of folders, after selecting from menu Word > File > Options > Advanced > General > File locations… (see Pic. 25). In other words, the user can change the location of the template files. I recommend keeping the default location, because then future scripting can be easier.



template folder A

template folder B

Pic. 10. Folders where Microsoft Word configuration is stored. [source: herein]

Tab. 5. Configuration of folders, File Locations. [source: herein]

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| folder short name aka file types (see Pic. 10) | behavior | files / content | default location | proposed location |
| User  templates | Stores so called user templates (compare Pic. 6): any Word configuration file (DOTM, DOTX). Upon pressing Attach… button or Add… button, content of this folder is shown. Files from that folder are not automatically loaded at startup, except the Normal.dotm. | Normal.dotm \*\* (1)  Styles\_ms.dotm(2)  MicrosoftDefaultBB.dotx (3) | C:\Users\<YourUsername>\AppData\Roaming\Microsoft\Templates | folder A see the column default location in this table |
| Startup | Automatically loads any Word configuration file (DOTM, DOTX) so called add-ins on time of Microsoft Word startup stored in this folder. All those files are loaded in read-only mode. Compare Pic. 10. | (5)  (6)  \* | c:\Users\<YourUserName>\AppData\Roaming\Microsoft\Word\startup\ | folder B  see the column  default location in this table |

\* Potentially in the same folder may be stored a DOTM file containing the Autotext definitions. I strongly advise replacing any Microsoft Word specific definition of Autotext except the building blocks with more general AutoHotkey solution. The further details could be found in (…)

\*\* This is a special, customized Normal.dotm file, containing special macros and altered settings related to default styles (see chapter 9.3).

The content of the specified folders A and B (files stored in those folders) is visible on time of Microsoft Word operation in menu Developer > Document Template (compare Pic. 11):

* 1. When the button Attach… is pressed, content of folder A is visible and one of the files from that folder can be selected.
  2. When button Add… Is pressed, content of folder B is visible and any of the files from that folder can be selected.

|  |
| --- |
| **🄸** Note |
| The menu section Developer > Document Template is surprisingly not visible by default, I mean, when Microsoft Word is installed. To enable it select File > Options > Customize Ribbon. Select from the right Main Tabs > Design. |



template **folder A**

template **folder B**

Pic. 11. Templates and add-ins ver. folder A and folder B. [source: herein]

* + 1. Location of theme file

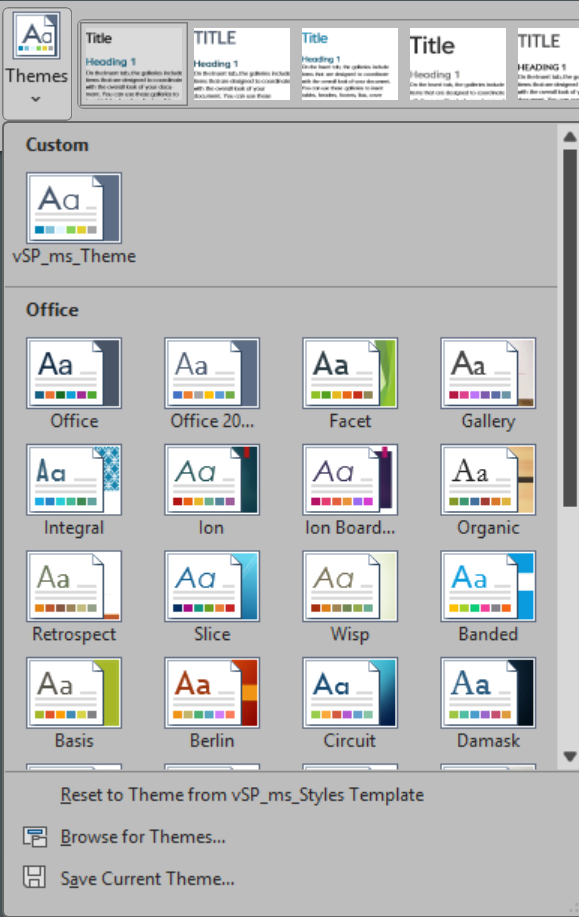
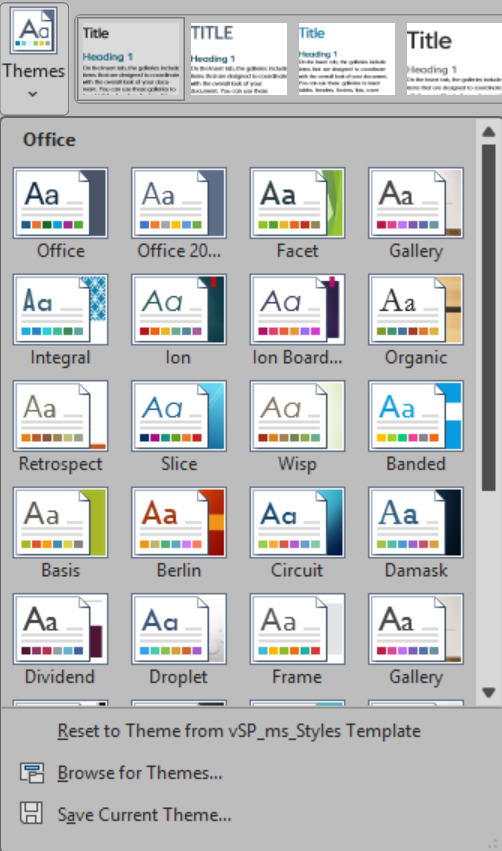
The customized theme files must be in the folder:

C:\Users\<YourUserName>\AppData\Roaming\Microsoft\Templates\Document Themes\

Content of this folder is connected with Microsoft Word GUI and menu Design > Themes:

* 1. If this folder is empty and there is no customized theme file, the menu Design > Themes contains only default themes, compare Pic. 8, the left part.
  2. If this folder is not empty and there is at least one customized theme file, the menu Design > Themes shows also the Custom section, compare Pic. 8, the right part.

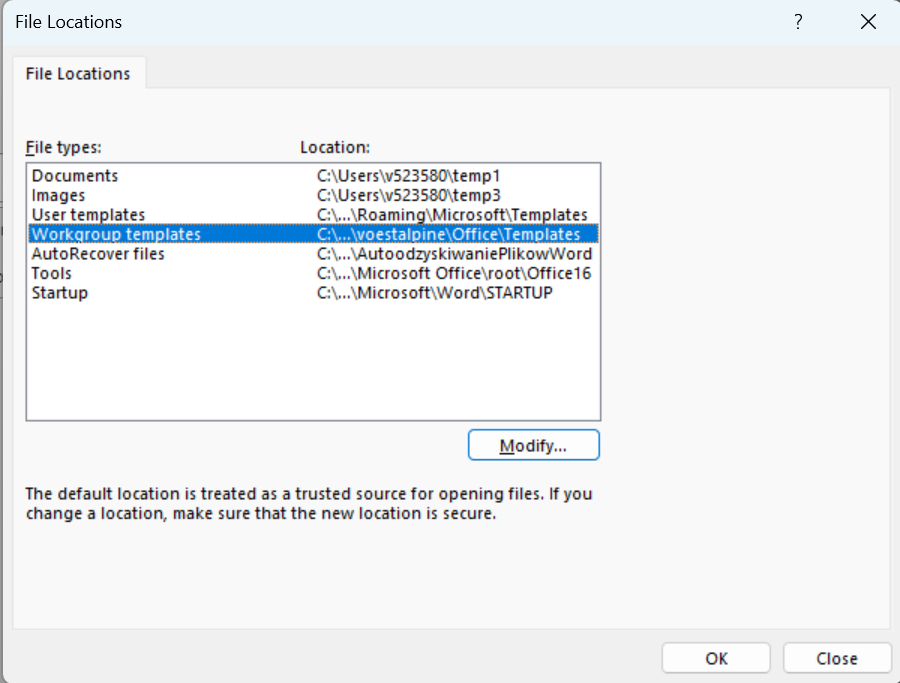
The folder of customized theme files cannot be changed from withing Microsoft Word configuration, as it used to be for template files (compare Pic. 10).



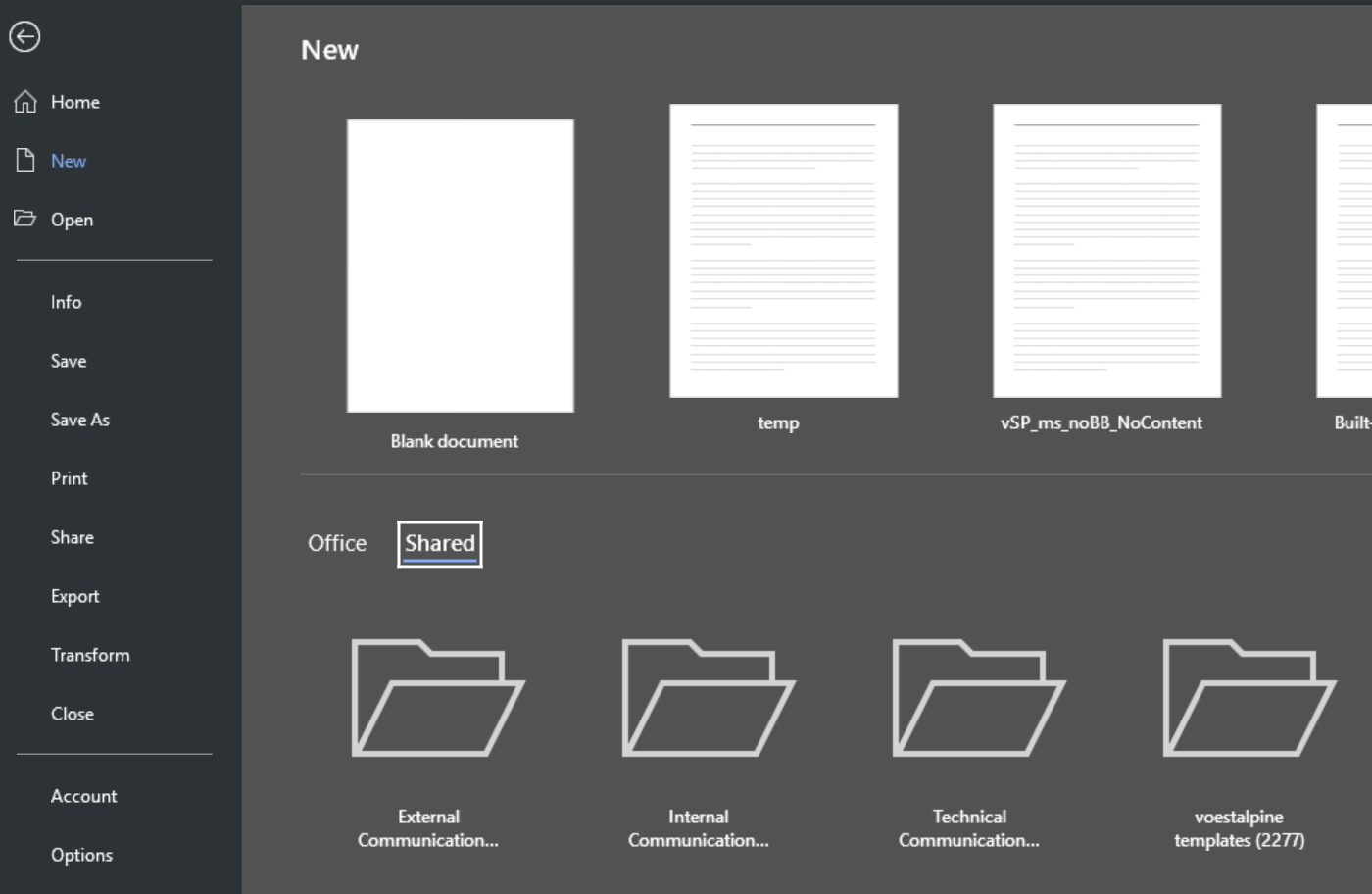
Pic. 12. Location of theme file. [source: herein]

* + 1. Menu file
  1. External configuration files and folders

After entering File > Options > Advanced > General and pressing File locations it is possible to set up the Workgroup templates folder where additional template configuration files are located (compare Pic. 21). If this folder, which by default is set to empty, is configured, then the user on time of creating a new file will have option to choose any of the templates from that folder: File > New > Shared (compare Pic. 22). It enables quicker choice of template from the predefined set.



Pic. 13. Workgroup templates. [source: herein]



Pic. 14. Access to workgroup templates. [source: herein]

* 1. Environment, backup operation files and folders

Further folder and file settings are available in Microsoft Word after selection of File > Option > Save. There is some inconsequence, as some settings are doubled and could be selected also from Word > File > Options > Advanced > General > File locations…, see chapter 3.2 for comparison:

* 1. AutoRecover file location.
  2. Default file location.

The Server drafts location is not relevant from my point of view, so I'll not touch it in this document.

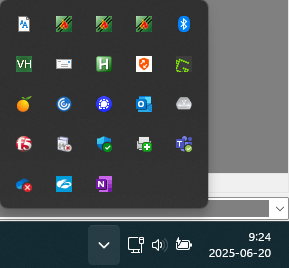
By the way I strongly recommend checking (☑) option Save AutoRecover. For more information regarding Backup see (…).



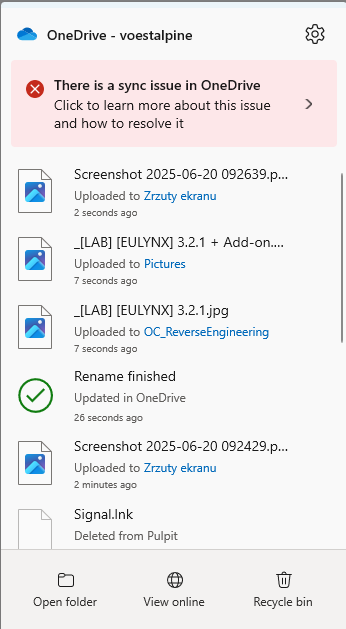
Pic. 15. File > Option > Save. [source: herein]

* 1. Environment, Microsoft Word and OneDrive

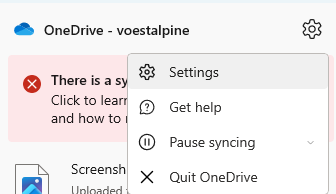
Because Microsoft Word could use (Microsoft) OneDrive as a file service, it is good to recognize relationship between these two applications. Observe green rectangle in the following pictures.



Pic. 16. Find OneDrive icon in SystemTray. [source: herein]



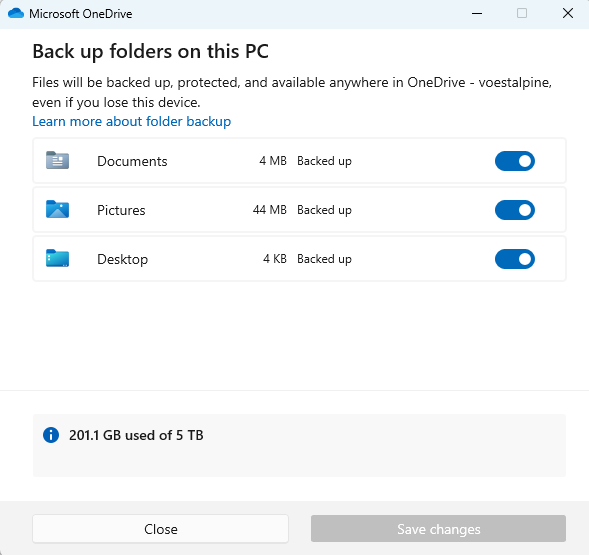
Pic. 17. Select settings (⚙ symbol). [source: herein]



Pic. 18. Select settings again (⚙ symbol). [source: herein]



Pic. 19. Select Manage backup button. [source: herein]



Pic. 20. Consider switching off. [source: herein]

|  |
| --- |
| **🄸** Info |
| I recommend switching off Pictures and Desktop folders. |

1. What are the template files?

There is no simple answer to that question. From my perspective I imagine them are like [containers](https://en.wikipedia.org/wiki/Container) or even [intermodal containers](https://en.wikipedia.org/wiki/Intermodal_container) in abstract form. They are like enclosures for holding different categories of information.

For me traditionally the “template” word has two meanings:

* + traditional, equal to stencil: set of rules dedicated e.g. to content,  
    you get a sheet of paper where only a few blank areas are ready to be filled in.
  + new, a set of configuration settings.

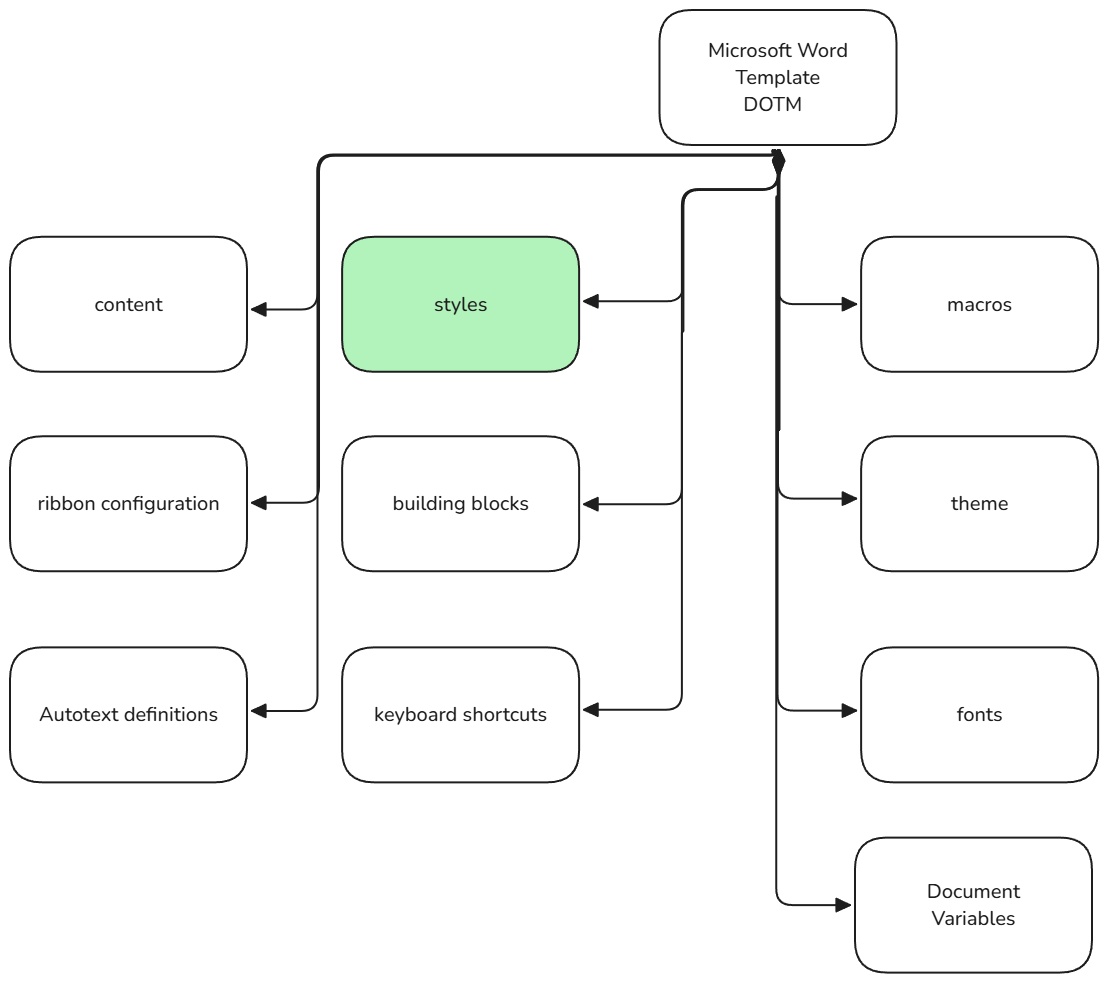
In this book the word template is applied in this second meaning. Actually, Microsoft Word template files can be both at the same time, but I decided to separate them from content.

The template files are binary. You can manually change the extension of such files to ZIP. Then you can see what information is stored inside. Don’t be surprised not seeing any of the subfolders or files from Pic. 21. The internal organization of files within ZIP folder isn’t so easy and straightforward.

* 1. What kind of information may be stored within a template file?

Legend:

* + green colors are marked with ever existing modules,
  + an arrow, which finishing end is diamond shaped and pointing end is a full black arrow, in SysML 1.1 notation should be read as “is made of”.



Pic. 21. Nazwy składowych konfiguracji, plik z rozszerzeniem DOTM. [source: MicrosoftWordTemplateConcept.excalidraw]

The DOTM template file from Pic. 21 is a container which contains just one module: styles. On the other hand, it can contain up to all 10 different information categories.

There is the second type of template file, DOTX. Files with this extension will not contain the macros category. Be aware that it is not enough just to change file extension. You must use Microsoft Word for that purpose and Save As function to change from one template category to another.”

* 1. Attachment relationship

What happens when you start the Microsoft Word application which is not yet altered or reconfigured? What you see on the screen is a blank file named Document1. Without explicit information to a user there is automatically created relationship called “attachment” between that file and the template file Normal.dotm. Thanks to that the file Document1 gets access to all modules of the Normal.dotm container. Again, by default the Normal.dotm contains just predefined styles, but it might contain much more. This attachment relationship lets the file Document1 to sore just content and separate it from other information. Compare the Pic. 8.

Please also note that by default both files: Document1 and Normal.dotm contain the same set of styles. Those styles are automatically copied from the Normal.dotm to Document1 on time of creation of Document1.

Microsoft Word

Document1

Normal.dotm

attachment

default styles

default styles

content

<empty>

content

<empty>

Pic. 22. Attachment relationship. [source: herein]

Let’s assume for a moment that we have already modified the template Normal.dotm by adding few macros, compare Pic. 8.

Microsoft Word

Document1

Normal.dotm

attachment

default styles

macros  
MyMacro1, MyMacro2

default styles

content

<empty>

content

<empty>

Pic. 23. Normal.dotm with macros. [source: herein]

Now the Document1 will have access to the macros MyMacro1 and MyMacro2 only if template file Normal.dotm is attached. If you send Document1.docx itself to any other user and she / he would like to run any of those macros, it won’t be possible. In other words, some categories of information are separated or accessible only if specific relationship is established.

On the other hand, all files which had attachment relationship to the Normal.dotm would have access to the macros.

Here comes my idea, that macros, if possible, should be stored in separate DOTM template file.

The attachment relationship is quite special. If the template is updated (e.g. new styles or macros), those changes can propagate to all documents attached to it, but only if special tick named Automatically update document styles is on (compare Pic. 11). The document stores the full path to the template. If the template is moved or deleted, Microsoft Word may show errors or revert to default styles.

* 1. What are the other relationships beside the attachment?

Besides the attachment relationship there is also add-in relationship. A single content document can have only one attachment relationship and numerous add-in relationships.

Document1

default styles

Normal.dotm

Styles\_ms.dotm

customized styles

customized styles

default styles

Macros\_ms.dotm

macros (main)

keyboard shortcuts

BB\_ms.dotm

building blocks

keyboard shortcuts

keyboard shortcuts

attachment

add-in

add-in

macros (special)

content <empty>

content <empty>

Pic. 24. My setup. [source: herein]

Please compare Pic. 28 and Pic. 23. For clarity the Pic. 28 do not contain the Microsoft Word frame. The Pic. 28 shows what happens when new Document1 is created with attachment relationship to Styles\_ms.dotm:

* + One-time automatic actions:
    - Copy of default styles from Normal.dotm to Document1.
    - Copy of customized styles from Styles\_ms.dotm to Document1.
    - Copy of content from Styles\_ms.dotm to Document1. Because the Styles\_ms.dotm doesn’t contain any content, it is not copied, but I just want to show what and from where is copied.

The Document1 has now access to all information categories from all three files: Styles\_ms.dotm, Macros\_ms.dotm, BB\_ms.dotm.



Pic. 25. Relationship with templates. [source: herein]

As you can see my templates do not store directly any content.

* 1. What does configuration "context" mean in Microsoft Word?

One of the quirks of Microsoft Word is (configuration) context. According to Microsoft there are following types of contexts:

1. Template or add-in.
2. Content file.
3. Application.

The context decides from where a configuration is loaded or where it is stored.

From the end user perspective, the meaning of context could be relevant. From maintainer / VBA programmer, understanding of this term and its consequences is crucial.

In particular, the third type, the application context, is a questionable idea, as some settings related to application context cannot be stored within a file, so it means they are not transferable. On the other hand, Microsoft Word has implemented dynamic application context switching. It means that depending on the current state of application a context menu available upon right mouse button click may look different or additional (usually in blue) menu ribbon sections may be displayed. Unfortunately, options to configure dynamic context menu available for user are limited to zero.

* + 1. When does a context matter?

Few examples (compare Pic. 28):

* 1. The keyboard shortcuts can be stored in a template file (DOTM), add-in file (DOTM) or content file (DOCX), although storing them in content file is for me questionable.
  2. Building blocks can be stored in a template file (DOTM) or add-in file (DOTM).
  3. Styles can be stored in a template file (DOTM) or content file (DOCX).
  4. Macros.

In other words, be careful where you store keyboard shortcuts, or building blocks, or styles, or macros.

From VBA developer perspective, whenever context matters, the CustomizationContext property in VBA code must be set. Possible values:

* + CustomizationContext = ActiveDocument.AttachedTemplate
  + CustomizationContext = Templates("TemplateProject")

In Microsoft Word VBA, the Templates collection does exist, and you can see its members in the Watches window. However, the confusion arises because VBA does not expose a Template object type that you can use directly in a Dim statement like:

Dim TemplateName As Template  ' ❌ This causes a compile error

If the same macro exists in different context, the AttachedTemplate prevails.

* 1. How do you make the attachment relationship between a content file and template manually?

This can be an implicit or explicit relationship:

* 1. implicit, as by default, if not customized in specific way (described in one of the following chapters) content file (DOCX or DOCM) will have automatically established relationship to Normal.dotm.
  2. Explicit, as user can change a default Normal.dotm to any other configuration file manually by entering ribbon menu Developer > Document Template and press the Attach… button to select it.

|  |
| --- |
| **🄸** Info |
| The menu section Developer > Document Template is surprisingly not visible by default, I mean, when Microsoft Word is installed. To enable it select File > Options > Customize Ribbon. Select from the right Main Tabs > Design. |

* 1. How the tick Automatically update document styles works?

Please look in Pic. 28 and Pic. 29 on time of reading this subchapter. To differentiate between ordinary file attachment and situation when the Automatically update document styles is active, I call it “constant attachment”: compare Pic. 28 and Pic. 24.

Document1

default styles

Normal.dotm

Styles\_ms.dotm

customized styles

customized styles

default styles

Macros\_ms.dotm

macros (main)

keyboard shortcuts

BB\_ms.dotm

building blocks

keyboard shortcuts

keyboard shortcuts

attachment

add-in

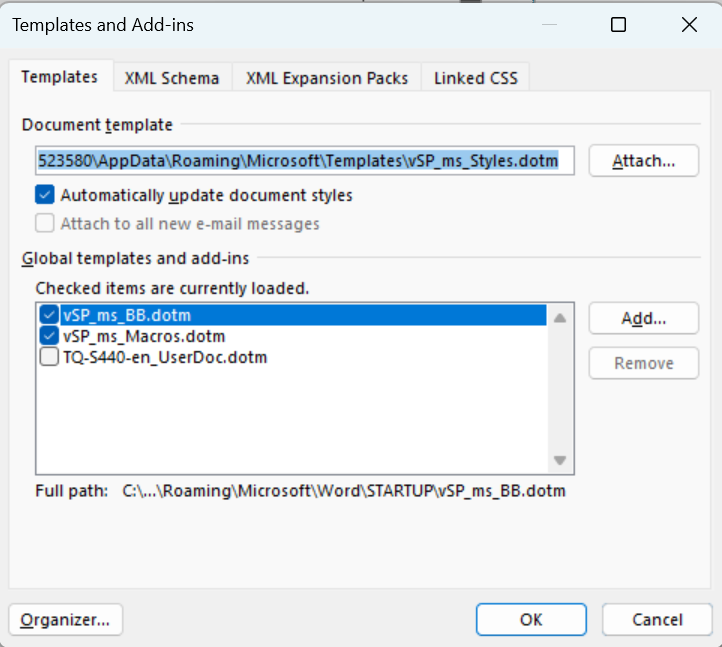
add-in

macros (special)

content <empty>

content <empty>

Pic. 26. Constant attachment. [source: herein]



Pic. 27. Automatically update document styles is on. [source: herein]

If the tick option Automatically update document styles is on (active, ☑) then each time you reopen your content file, styles from the Document template are automatically (silently) overwritten with the freshest definitions from that template.

If this tick is activated for the very first time, styles from the Document template are automatically (silently) copied or overwritten from that template.

Why is this function there? I have no clue. In my opinion the idea of automatic (silent) update of styles definition is quite strange, as users don’t get feedback about what styles and when have been updated. I don’t like it, and I don’t recommend enabling it. My preference is to copy styles definitions consciously. The set of styles definitions should remain stable, so there is most of the time no reason to update them.

* 1. Can I use a template file as stencil or content template?

Any template file (DOTM or DOTX) may contain content. Then if the user starts Microsoft Word or creates a new document in any other way, the newly created Document1 file will contain the same content as the template file. It is tempting to store content within a specific template file. I prefer another approach.

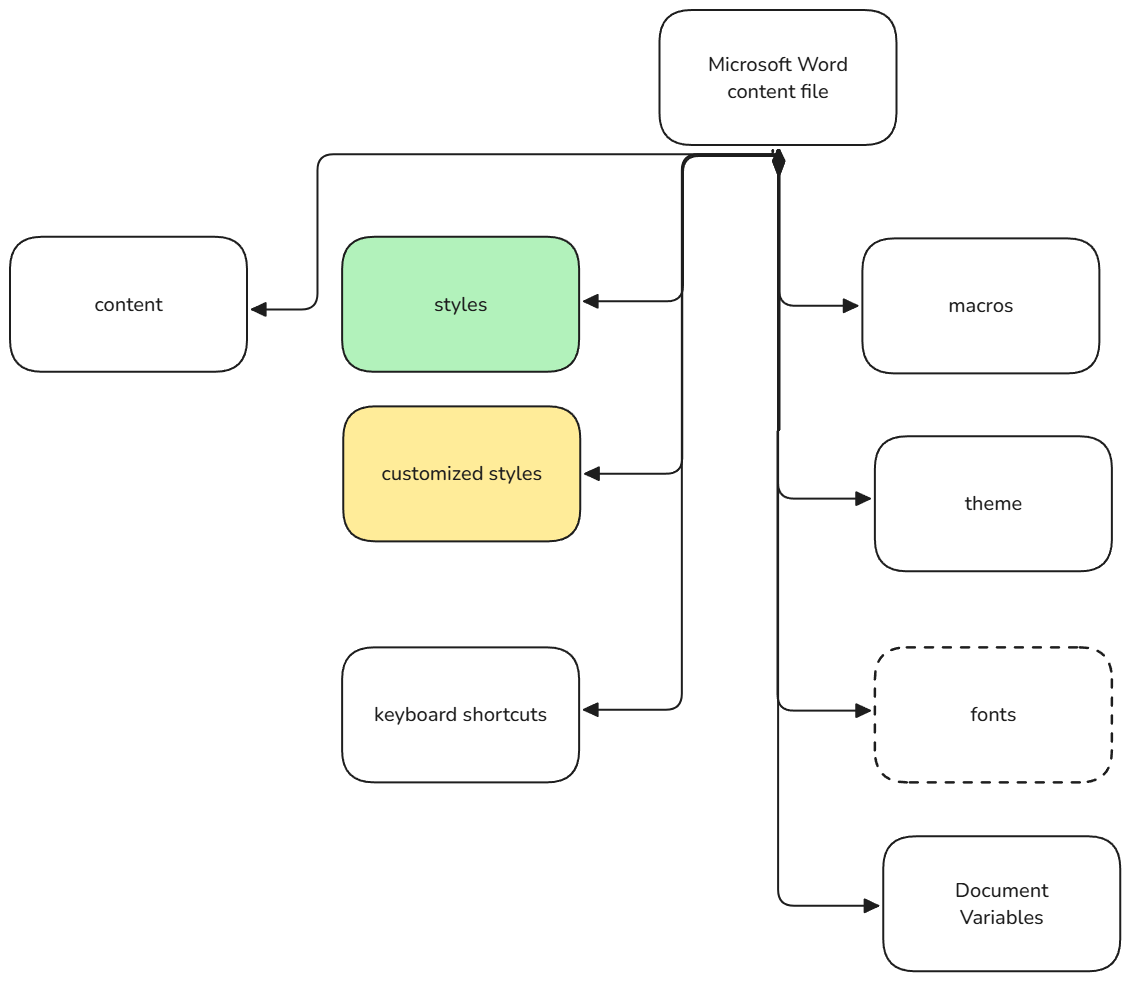
Any file, e.g. DOCX content file, can be used for the purpose of storing a stencil like template content. Just open such a file and save it under a new name, next edit it and alter these parts of content which you like.

If you wish to create a set of repeatable content, like in a stencil, use for that purpose building blocks (…). Compose a new content file DOCX out of existing building blocks. This is the recommended and preferred solution. I opt for clear separation of configuration and the content.

Nevertheless, a choice is yours. Remember that in chapter 3 I already briefly touched different strategies of separation configuration and content.

1. What are the theme files?
   1. What are the menu files?
2. Content files

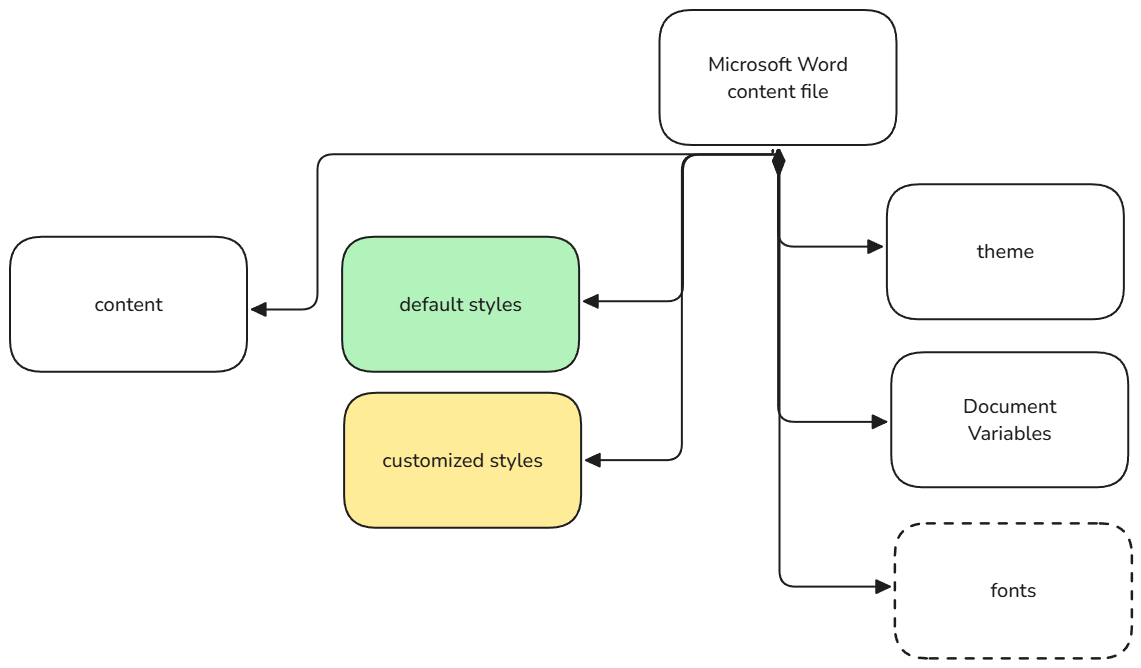
For comparison, please see in Pic. 28 the information categories which can be stored within a content file.



Pic. 28. Information categories, content file. [source: MicrosoftWordTemplateConcept.excalidraw]

Remarks:

* + The green color represents default and mandatory information category, styles from Normal.dotm.
  + The yellow color represents optional customized styles.
  + Macros can be stored only in DOCM files. The type of content file must be changed manually from DOCX to DOCM to save macros in a content file. Such files are potentially a security threat, so security settings can be important in such cases. I don’t recommend storing macros within content files.
  + Fonts are stored within content files only on purpose. It can be necessary if you want to copy your content file to another computer which does not have access to your set of fonts. Embedding fonts was discussed in the chapter 11.4.
  + Keyboard shortcuts may apply only to styles and internal Microsoft Word commands (not to macros). I do not recommend storing any shortcuts within content files.
  1. Recommended information categories



Pic. 29. Recommended information categories. [source: MicrosoftWordTemplateConcept.excalidraw]

Remarks:

* + The green color represents default and mandatory information category from Normal.dotm.
  + The yellow color represents customized styles from Styles\_ms.dotm.
  + Fonts are optional and most of the time are not stored within a content file. Embedding fonts was discussed in the chapter 11.4.

As one can see there is much more than just content within such a file.

* 1. What happens to formatting if I move my content file to another computer?

Nothing wrong or surprising should happen unless you change the attached template file.

Remember about fonts. If destination computer do not have installed the same set of fonts as source computer, then the destination file will look differently, precisely speaking specific fonts will be replaced with fonts from .

1. Configuration of Microsoft Word Graphical User Interface

In this chapter I focus on specific aspects of Graphical User Interface, such as:

* 1. ribbon menu,
  2. status bar,
  3. panes.
  4. Design of Microsoft Word GUI

One of the unique features of Microsoft Word is hidden under horrible WYSIWYG acronym, which means What You See Is What You Get. In other words, Microsoft Word text or content processor can show on a screen the actual size, look and feel of a document, as it would look like upon printing.

The basic size of a paper sheet, at least in Europe, is A4:

a\_A4 = 210 mm

b\_A4 = 297 mm.

The very first size of computer display, which fits to this size is 24'' (inches), what means that its diagonal is equal to

diagonal = 24'' ≈ 24 · 25.4 mm ≈ 609.6 mm = c

By strange coincidence I have in my office exactly 24'' [physical display](https://www.displayspecifications.com/en/model/f79ee0e). By simple experiment I placed 2× sheets of A4 paper on a display, compare Pic. 32 and Pic. 37.



Pic. 30. 24'' Display and 2× A4 blank pages, side to side. [source: IMG\_2035\_24Display\_BlankPage.HEIC ]



Pic. 31. 24'' Display and 2× A4 blank pages, moved to the left. [source: IMG\_2034\_24Display\_BlankPage.HEIC ]

Following my 24'' display specification:

a\_display = 301 mm > b\_A4 by 4 mm

b\_display = 533.5 mm > 2 · b\_A4 by 113.5 mm:

533.5 mm − 2 · b\_A4 = 533.5 mm − 2 · 210 mm = 113.5 mm

Results:

* 1. It is possible to keep 2× A4 sheets of paper on the 24'' display.
  2. If you wish then to keep any menu "traditionally", I mean horizontally at the top, you are left with just 4 mm.
  3. But you have 113.5 mm which is 38.2% of horizontal for vertical menu(s).

So maybe it would be more reasonable to keep all menus, e.g. operating system and application, like Microsoft Word vertically? Then why can't a Microsoft Word ribbon menu be moved from its top horizontal position? Hopefully now you see more clearly that drawback.

* 1. Ribbon menu

The name ribbon for top menu was introduced by Microsoft. It has a big advantage that can be quite widely configured. Unfortunately, it cannot be unlinked from the top part of a screen, neither moved to either screen side.

Ribbon menu configuration can be exported / imported to a separate file with file extension .exportedUI. Once such file is imported, it is no longer updated upon changes added by user. You've been warned.

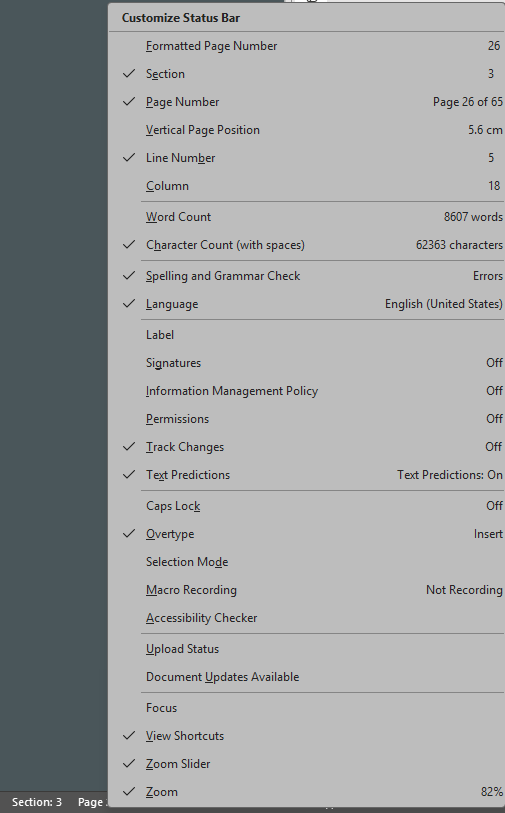
* 1. Status bar

Settings of status bar are not available from VBA code. They are not saved besides application context.

To enter settings of status bar move mouse pointer over bottom left part of Microsoft Word GUI and right click. I strongly recommend copy settings from the Pic. 19 or just pure experimenting.

There are two handy keyboard shortcuts to toggle specific options, which state is reflected at status bar:

* 1. Insert (INS): toggles insertion mode, aka text replacement mode of typing.
  2. Ctrl + Shift + E for toggling tracking changes.



Pic. 32. Example settings of status bar. [source: herein]

* 1. Panes

A pane in Microsoft Word is a small, usually dockable subwindow. As working with panes is user specific, I didn't prepare macros to enable panes. Below you can find few panes and shortcuts which I find in particular handy. Most of the time I switch / open / close them dynamically with keyboard shortcuts.

Tab. 6. Selected Microsoft Word panes. [source: herein]

|  |  |  |  |
| --- | --- | --- | --- |
| pane name | keyboard shortcut | pane content | comment |
| Navigation > Headings | special, supported by macro:  Alt + Ctrl + H | Conjugated to style levels window | Dockable. Extremely useful to change document structure / move content. |
| Reveal Formatting | built-in: Shift + F1 | Full formatting, including font, paragraph, list, table, cell, section | The only one useful tool to check content formatting. |
| Styles | built-in: Ctrl + Alt + Shift + S | List of few types of styles, access to Manage Styles | No comment. |

1. Running Microsoft Word

Following the principle of simplicity, and in consequence of work invested in design and development of dedicated configuration files, I want to:

* + Give the user a conscious choice if she / he wants to use the default Microsoft Word configuration or mine.
  + If mine configuration is selected, limit number of styles, building blocks and even macros and shortcuts to those which I have designed and developed.

Rozważam następujące tryby pracy:

* 1. Domyślny: edycja (treści) dokumentu.
  2. Konfiguracyjny: edycja plików konfiguracyjnych.
  3. Tryb pracy edycja dokumentu, scenariusze

Cel: chcę zautomatyzować proces dołączania składowych konfiguracyjnych.

Założenia:

* 1. Każdy nowo tworzony plik (nazwa domyślna: Document1.docx) powinien automatycznie uzyskać dostęp do plików:
     1. ze specjalną konfiguracją styli   
        plik: Styles\_ms.dotm,
     2. ze specjalnymi blokami konstrukcyjnymi   
        plik: BB\_ms.dotm,
     3. ze specjalnymi makrami  
        plik: Macros\_ms.dotm.
  2. Makra powinny w formie dialogu z użytkownikiem (pytanie / odpowiedź) zapewnić dołączenie plików:
     1. motywu (ang. Theme)  
        plik: Theme\_ms.thmx,
     2. przykładowej treści,
     3. skrótów klawiaturowych,
     4. ustawień konfiguracyjnych aplikacji Microsoft Word.

Rozważam następujące scenariusze:

* 1. Uruchomienie tylko Microsoft Word.
  2. Uruchomienie Microsoft Word poprzez wybór istniejącego pliku.
  3. Uruchomienie Microsoft Word poprzez wybór pliku szablonu.

Wynik dla pierwszego punktu założeń będzie podejrzeć wywołując dla nowo utworzonych plików z poziomu menu: *Developer → Document Template* (patrz Pic. 13).

Scenariusz 1.

Microsoft Word domyślnie, tj. bez naszej konfiguracji, po uruchomieniu otwiera plik Document1 i dołącza do niego plik konfiguracyjny Normal.dotm. Po wprowadzonych zmianach Microsoft Word po uruchomieniu otworzy plik Document1, a użytkownikowi zostanie zadane pytanie, czy ten chce dołączyć wskazany plik konfiguracyjny ze stylami, a same style zostaną przekopiowane do tego pliku. Następnie otrzyma serię kilku pytań dotyczących ustawienia środowiska Microsoft Word oraz wstawienia do pliku przykładowej treści.

Scenariusz 2.

Microsoft Word może zostać otwarty poprzez wywołanie istniejącego pliku treści (np. DOCX), np. poprzez dwukrotne kliknięcie na ikonie lub/i nazwie tego pliku. Domyślnie, tj. bez naszej konfiguracji, nic już więcej się nie dzieje. Jeżeli do tego pliku został wcześniej dołączony specjalny plik konfiguracyjny (DOTM), to po uruchomieniu użytkownik otrzyma serię kilku pytań dotyczących ustawienia środowiska Microsoft Word.

Scenariusz 3 (rzadziej praktykowany).

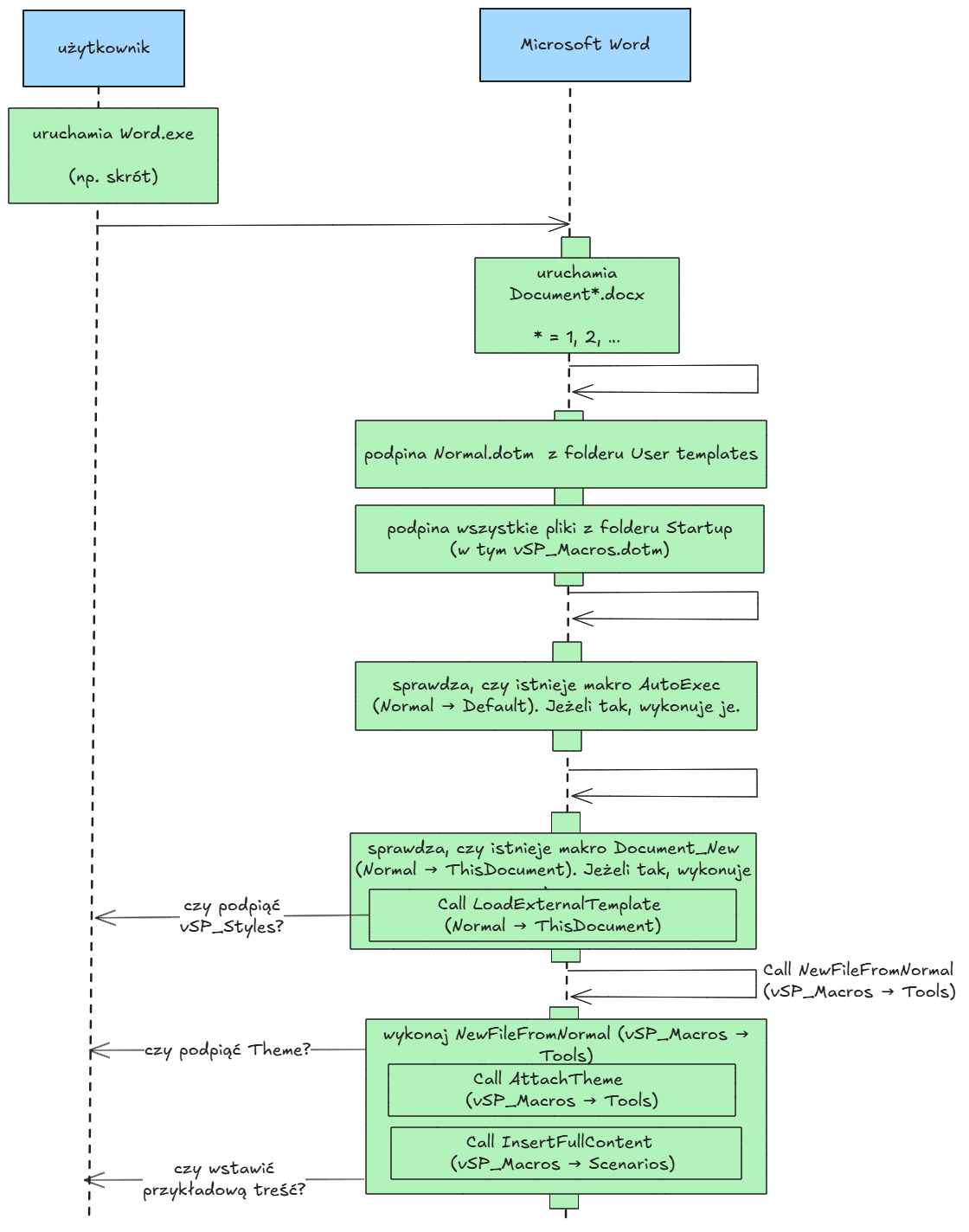
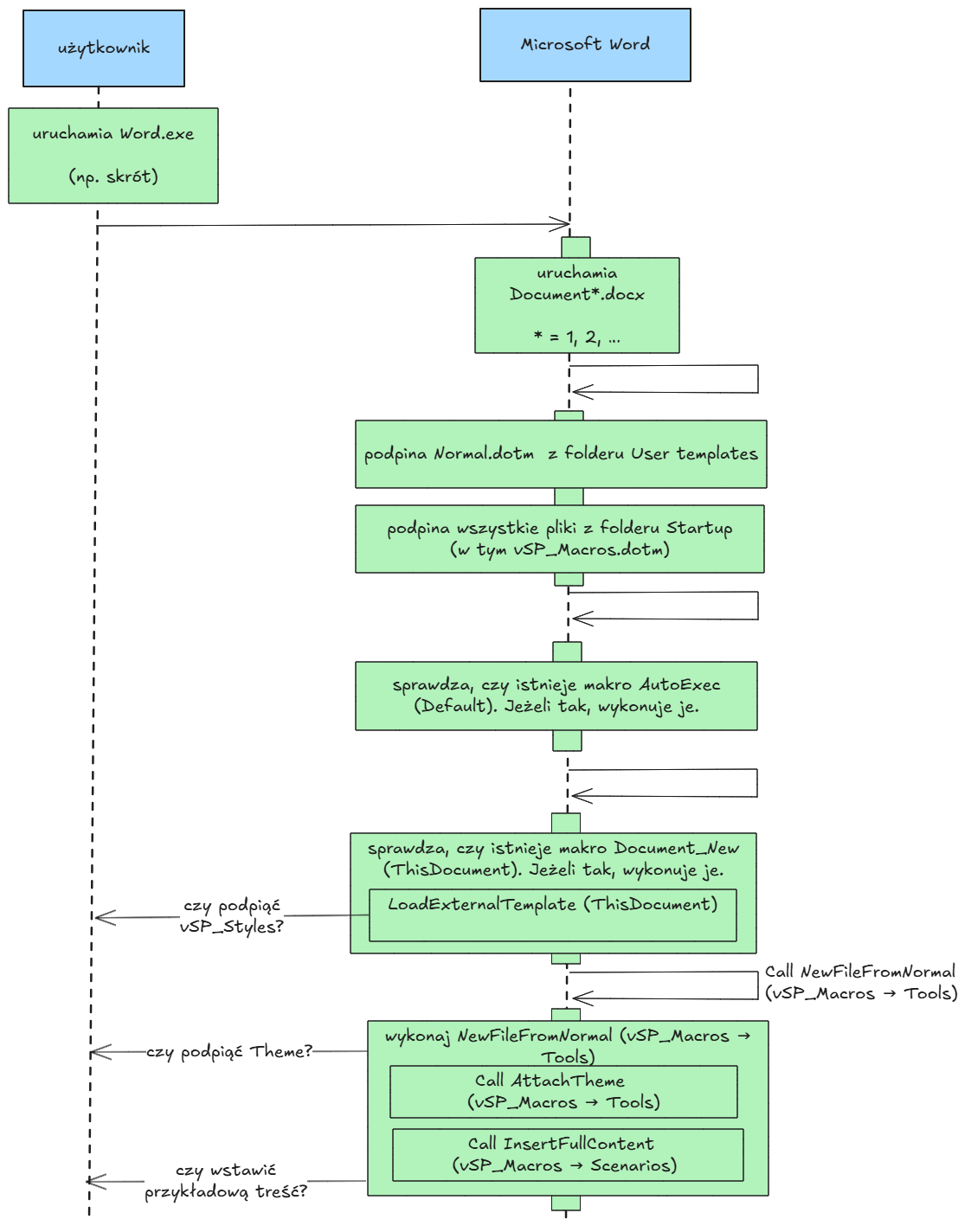
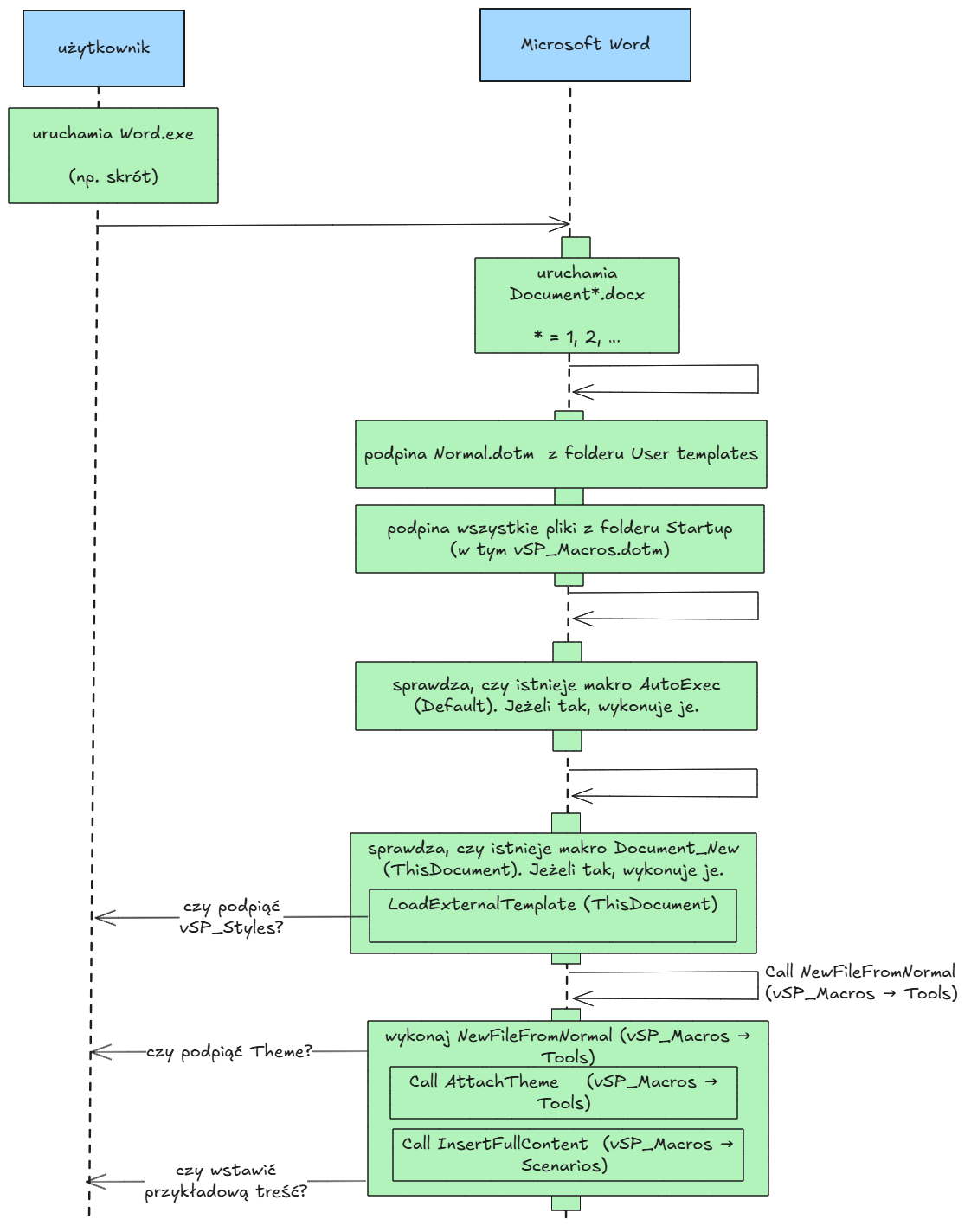
Utworzenie nowego pliku poprzez wywołanie pliku konfiguracyjnego (DOTM), np. poprzez dwukrotne kliknięcie na ikonie lub nazwie tego pliku. Domyślnie, tj. bez naszej konfiguracji, zostanie utworzony plik Document1 i zostanie do niego automatycznie dołączony ten plik konfiguracyjny (DOTM). Jeżeli jednak był to specjalnie przygotowany plik konfiguracyjny, to ponownie użytkownik otrzyma kilka pytań, które umożliwią zmianę ustawień środowiska Microsoft Word oraz wstawienie do pliku przykładowej treści.

Ustawienia wspólne dla wszystkich scenariuszy:

Użytkownik będzie miał dostęp do szeregu makr i bloków konstrukcyjnych, do pliku szablonu i do pliku nadającego określone ustawienia GUI.

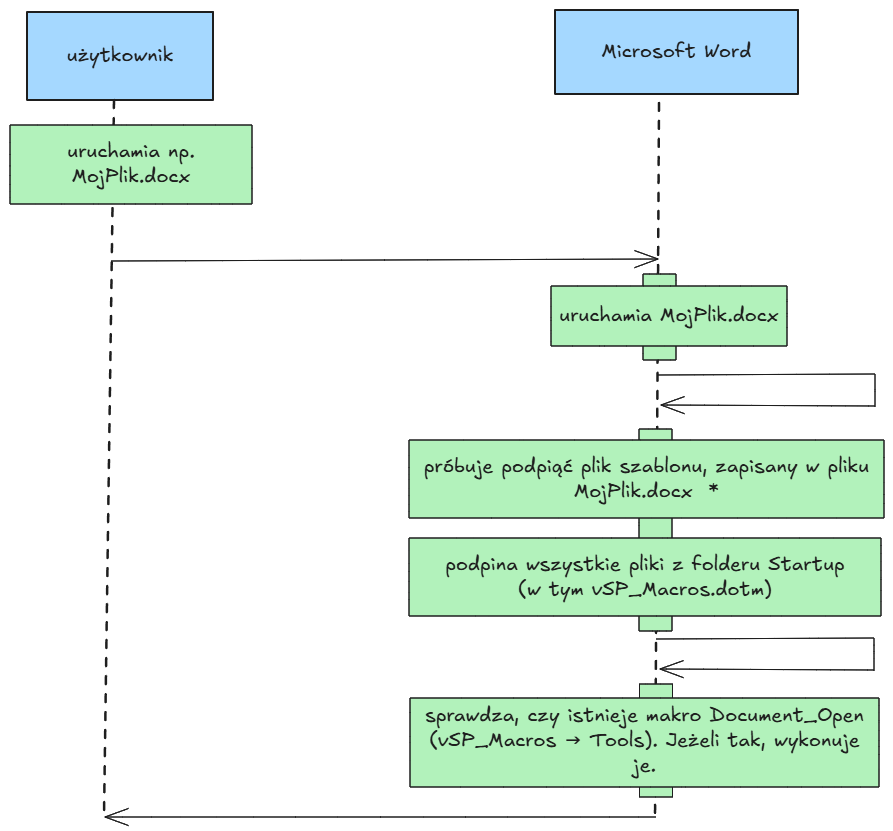
Oprócz wskazanych powyżej celów doraźnych, do zrealizowania są także cele dodatkowe:

* + Jawne informowanie użytkownika o działaniach dotyczących pracy i konfiguracji Microsoft Word.
  + Zwiększenie efektywności pracy poprzez ułatwienie dostępu do wybranych funkcji Microsoft Word.
    1. Uruchomienie samej aplikacji Microsoft Word



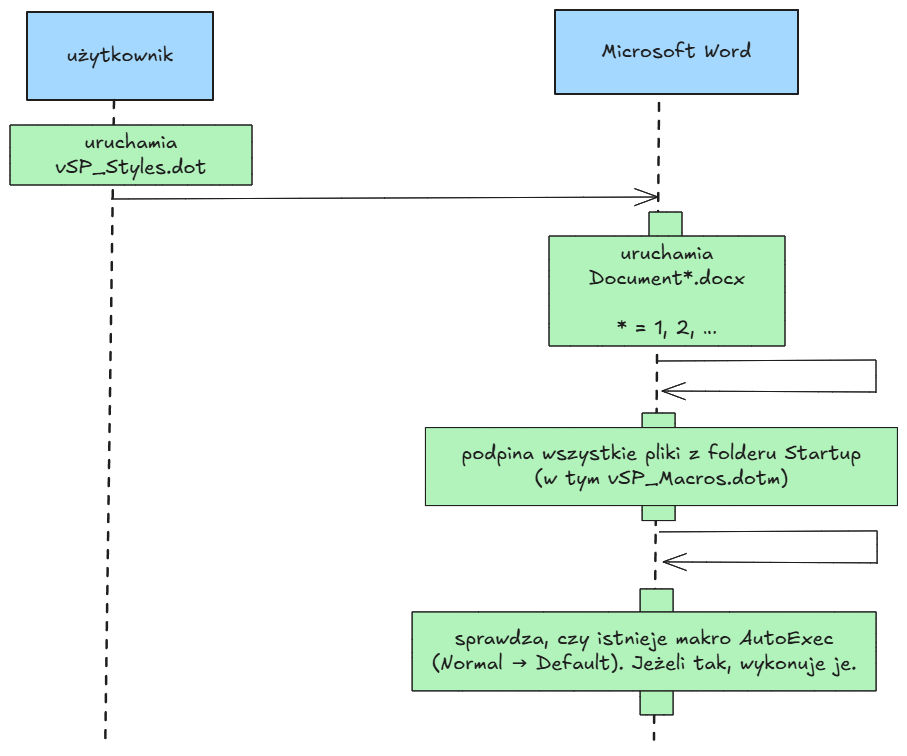
Pic. 33. Scenariusz 1, sam Microsoft Word. [source: 20250715\_MicrosoftWordScenario1.excalidraw]

* + 1. Uruchomienie Microsoft Word poprzez wybór istniejącego pliku



Pic. 34. Scenariusz 2, uruchomienie istniejącego pliku. [source: 20250715\_MicrosoftWordScenario2.excalidraw]

* + 1. Uruchomienie Microsoft Word poprzez wybór pliku szablonu

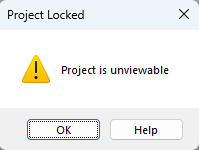


Pic. 35. Scenariusz 3, wybór szablonu. [source: 20250715\_MicrosoftWordScenario3.excalidraw]

Uwagi:

* 1. Założenie projektowe było takie, że plik Styles\_ms.dotm ma zawierać tylko style (ma nie zawierać makr).
  2. Możliwe, że warto zmienić DOT → DOTM, by wyposażyć Styles\_ms.dotm w makro Document\_New, które załaduje np. wstawianie przykładowej treści.
  3. Tryb pracy edycja plików konfiguracyjnych

| **⚠** Ostrzeżenie |
| --- |
| Nie można edytować treści makr wywołując edytor VBA dla aktualnie otwartego pliku DOCX. Plik zawierający makra (Macros\_ms.dotm) jest dołączany jako tzw. rozszerzenie (ang. add-in) do aktualnie otwartego pliku DOCX w trybie tylko do odczytu (ang. read-only). Z tego powodu przy próbie otwarcia pojawia się komunikat (…). |



Pic. 36. Example. [source: ]

Należy osobno otworzyć plik zawierający makra (Macros\_ms.dotm) poprzez File > Open. Co ciekawe edytor VBA pozwala w takiej sytuacji na sekwencyjne wywoływanie zmodyfikowanego kodu w aktualnie otwartym pliku DOCX. Nie jest przy tym ważna kolejność otwierania plików DOCX i DOTM, tzn. na przykład można najpierw otworzyć plik DOCX, a potem plik Macros\_ms.dotm.

* 1. Normal.dotm

1. Microsoft Word processor workflows

There are two workflows:

* 1. content edition,
  2. content formatting.

On time of content edition it is good to keep content unobstructed with any unprinted characters, sizes, color of page background etc. Opposite to that on time of content formatting it is good to see every supporting character or text processor feature which could support process of content formatting.

To make it possible I've prepared macro which is by default connected to F4 keyboard key. It works as a toggle between the following modes:

* 1. Visible all unprinted characters (formatting marks), hidden text, field codes, bookmarks, text boundaries, crop marks.
  2. Background color toggled to grey color (RGB = 219, 219, 219).
  3. Invisible all printed characters (formatting marks), hidden text, field codes, bookmarks, text boundaries, crop marks.
  4. Background color toggled to default one (no color = white).

Macro name: ToggleSpecificFormatting in module Tools.

Unfortunately, Microsoft Word by itself scatters all those options related to visibility of characters and page background all over its menus.

1. Informacje składowe wykorzystywane przez Microsoft Word

Wstęp (…).

* 1. Content

In general the configuration files (DOTX or DOTM) may contain content. It could be potentially useful to a user to get a kind of ReadMe content within such a file. For example I can imagine that after opening e.g. C\_F\_Macros reader would see information explaining how and where to store such a file. Or even upon opening of such file a macro Document\_Open can ask a user if to insert such information into document body.

Some configuration files could call specific macros which produce automatically generated lists or updated list referring to their actual content. For example, the mentioned C\_F\_Macros can generate list of available macros and insert it into a body of that file.

At the moment such functions are not implemented. As a rule of a thumb, if you wish to see a content of specific file, you need to run a macro which name starts with prefix Count. Again, if you wish to know what macros are stored within the configuration file C\_F\_Macros you may run the macro CountSubsInTemplate and look into a content of generated text file CountSubsInTemplate.txt.

Because of the fact it is possible to store content within a configuration file, it is very tempting to keep configuration and content always together in just one file. Does it matter today that such a file would be bigger in size? To be considered…

* 1. Style (C\_F\_Styles.dotm)

Definicje styli:

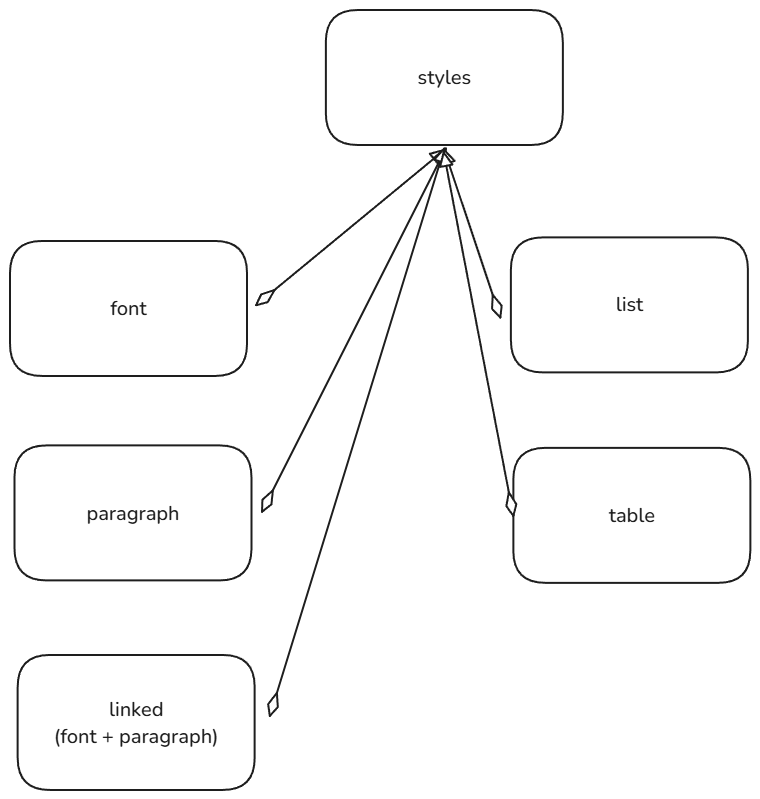
* 1. Nie są objęte prawem autorskim. Innymi słowy nie powinny zawierać nic, może poza czcionką, co byłoby w sensie prawnym utworem / wymagało ochrony.
  2. Powinny być spójne w różnych plikach szablonów, niezależnie od ich wersji językowej (np. TQ-S440-en\_UserDoc.dotm i TQ-S440-pl\_DokUzyt.dotm powinny mieć ten sam zestaw takich samych szablonów).

Ogólne zasady. Wszystkie nowe style powinny dać się odróżnić od styli wbudowanych. Dobrą zasadą jest dodanie przyrostka, np. „ms”. Złą zasadą będzie dodanie przedrostka, ponieważ może to utrudnić odnalezienie właściwego stylu.

* + 1. Why to make this file?

(…)

* + 1. How to make this file?
    2. Jak wygląda klasyfikacja typów styli?

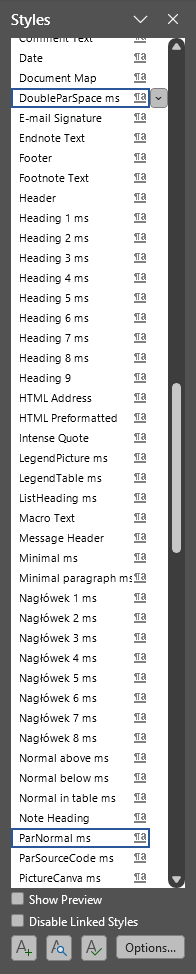


Rys. 1. Klasyfikacja typów styli. [source: MicrosoftWordTemplateConcept.excalidraw]

Legenda:

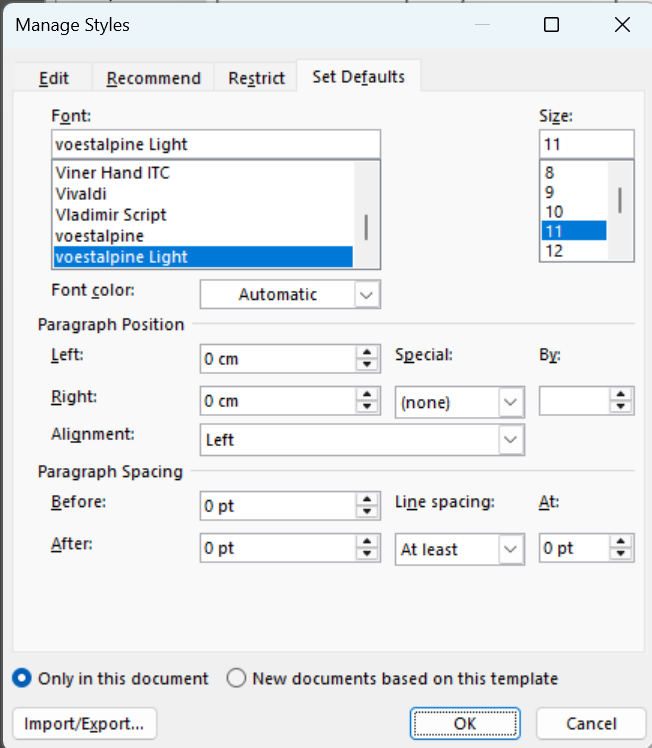
* + strzałkę, której tylec zakończony jest konturem rombu, a grot ma postać ostrą, bez wypełnienia, czytamy „jest podtypem”.
    1. Quirks of default styling "Set Defaults"

For each file (no matter if it is configuration file or content file) there is available managing of styles. By default you get there by calling the Styles pane by keyboard shortcut Ctrl + Shift + Alt + S and then selecting small icon Manage Styles (green in the following picture).



Pic. 37. Manage Styles. [source: herein]

In the Manage Styles window there is the tab Set Defaults.



Pic. 38. Set Defaults. [source: herein]

Surprisingly:

* 1. There is no way to set up those settings from VBA code, at least on 2025-07-25. In other words you must change them manually and cannot automatize this task.
  2. Those settings have influence to subtle formatting of the file, such as header and footer distances between paragraphs, vertical distances within tables, no name a few.
  3. For existing file, it is crucial to keep settings as specified in Pic. 38.
  4. For newly created file, based on the recommended template (Styles\_ms.dotm), settings of the Set Defaults are imported from Normal.dotm > Set Defaults. In other words: you must either edit your Normal.dotm > Set Defaults once (keep it mind), either edit each newly created content file to check if settings of Set Default as shown in Pic. 38.

This is one of the biggest and the most annoying quirks in Microsoft Word. You've been warned.

* + 1. Paragraph settings, line spacing

For some crucial styles (e.g. ParNormal ms, Heading 1 ms, Heading 2 ms, Heading 3 ms) in order to keep need and compact formatting, specific Exactly line spacing was applied.



Pic. 39. Line spacing: Exactly. [source: herein]

Comments:

* 1. At value follows font setting for the style, e.g. for ParNormal ms at is 14 pt.
  + The line spacing will *not* dynamically adjust. If you have text or objects on a line that are larger than the "Exactly" value you've set, they will be **clipped or overlap** with the text on adjacent lines. Surprisingly superscripts or subscripts are not clipped, e.g. H2O.
    1. Jakie są najważniejsze skróty klawiaturowe związane z wykorzystaniem styli?

|  |  |  |
| --- | --- | --- |
| Znaczenie | Skrót klawiaturowy | Kiedy to się może przydać? |
| Przywróć domyślne ustawienia stylu dla akapitu | Ctrl + W | Jeżeli chcesz zastosować ponownie formatowanie stylu do akapitu i chcesz, by było to dokładnie takie formatowanie, jak w stylu zapisanym w pamięci (bez modyfikacji dodanych przez użytkownika) |
| Przywróć domyślne ustawienia stylu dla czcionki | Ctrl + Space | Jeżeli np. kilka wyrazów zostało sformatowanych z przekreśleniem. Wystarczy je zaznaczyć, użyć podanego skrótu i w ten sposób przywrócić domyślny styl "ParNormal ms”. |
| Podgląd pełnego formatowania | Shift + F1 | Tylko to okienko pokazuje style złożone (dla list i tabeli). |

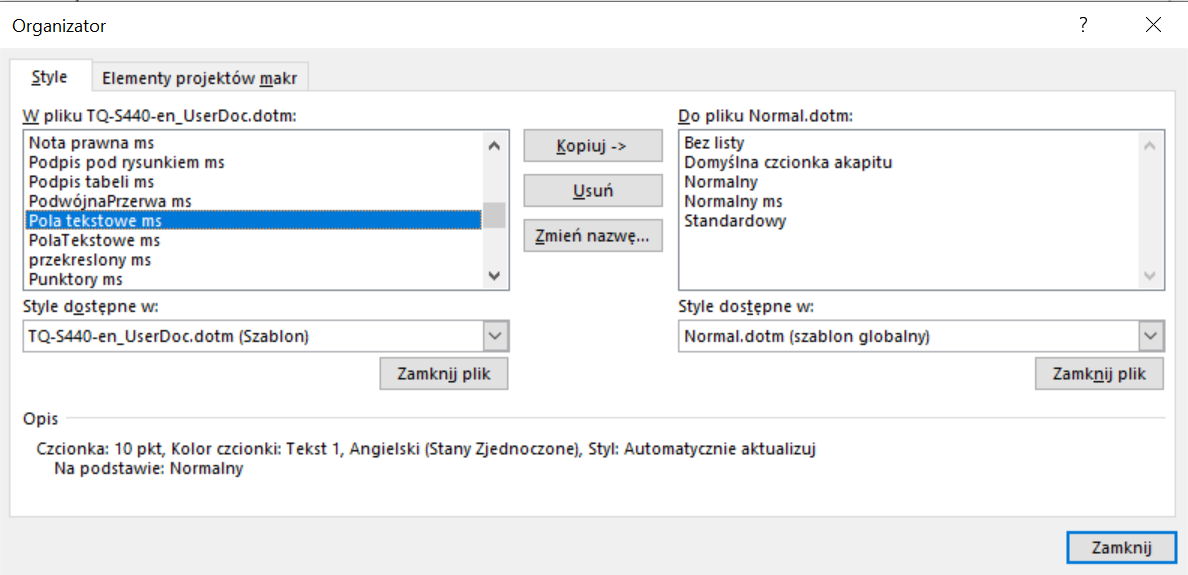
* + 1. Jak usunąć zbędne style z szablonu?

Posłużyć się „Organizatorem” styli. Menu „style ms” → „organizator styli”.

Otwórz plik szablonu (DOTM).

Kliknij w "organizator styli".

Ręcznie usuń zbędne style z pliku szablonu. Przydałoby się makro.



Oprócz styli „ms” mają prawo zostać tylko style widoczne po prawej stronie, czyli:

Bez listy.

Domyślna czcionka akapitu.

Normalny.

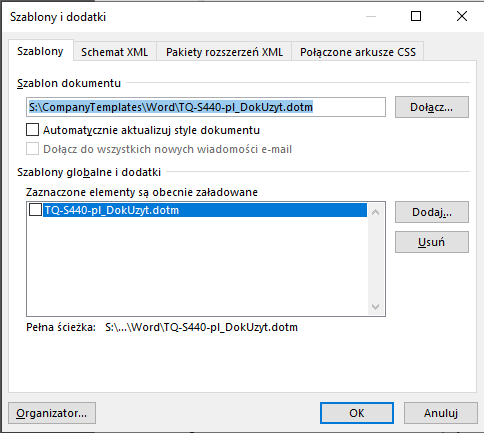
Standardowy.

* + 1. Czy do pliku DOCX da się dołączyć szablon bez automatycznego kopiowania styli?

Tak, da się:

do przeniesienia styli posłużyć się „organizatorem” styli,

podłączyć szablon bez zaznaczania opcji „Automatycznie aktualizuj style dokumentu”.



Dołączenie szablonu bez przekopiowywania styli.

* + 1. Dlaczego ktoś może chcieć to zrobić?

Ponieważ kopiowanie styli z plików szablonów (DOTM) do plików treści (DOCX) nie działa prawidłowo.

* + 1. Jak sprawdzić, czy w dokumencie prawidłowo wykorzystane są style z szablonu?

Wszystkie style typu „akapit” oraz połączone typu „akapit i znak” w naszych szablonach oparte są na stylu *Normalny ms*, który jest aliasem stylu wbudowanego *Normalny*. Dzięki temu zabiegowi możliwe jest sprawdzenie poprawności budowy bloków konstrukcyjnych jak i samych styli poprzez zmianę czcionki w stylu „Normalny ms” Calibri na jakąś radykalnie się różniącą, na przykład Excalifont-Regular.ttf. Po przestawieniu czcionki na Excalifont-Regular.ttf wszędzie tam, gdzie pozostała czcionka Calibri coś jest nie tak i wymaga poprawienia. Po poprawieniu ponownie przestawiamy czcionkę w stylu na *Normalny ms* na Calibri.

* + 1. Czy gdzieś można zobaczyć, jak wyglądają wszystkie style znajdujące się w szablonie?

Tak, w samych plikach szablonu. Pliki szablonu same dokumentują zawartą w nich treść. Stąd znajduje się w nich rozdział pokazujący wygląd styli oraz kontekst ich użycia.

* + 1. Projekt stylu „Normalny ms”

Największy wpływ na wygląd dokumentacji ma styl „Normalny ms”.

Wymagania zewnętrzne: [Web Content Accessibility Guidelines (WCAG)](https://en.wikipedia.org/wiki/Web_Content_Accessibility_Guidelines).

Styl jest przystosowany do czytania na ekranie komputera. Dlatego nie ma wcięcia specjalnego z prawej i jest wyrównany do lewej, a nie wyjustowany.

Typ stylu: akapit.

Na tym stylu są oparte wszystkie pozostałe style "ms".

* + 1. Style nagłówków

Tab. 1. Najważniejsze wartości definiujące style nagłówków.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| styl | akapit przed [pkt.] | akapit po [pkt.] | specjalne wysunięcie [cm] | wcięcie z lewej [cm] | pogrubienie | czcionka |
| Nagłówek 1 ms | 24 | 6 | 1,25 | 0 | nie | 14 |
| Nagłówek 2 ms | 18 | 6 | 1,25 | 0 | nie | 14 |
| Nagłówek 3 ms | 18 | 6 | 1,4 | 0 | nie | 12 |
| Nagłówek 4 ms | 18 | 6 | 1,6 | 0 | nie | 10 |
| Nagłówek 5 ms | 18 | 6 | 1,78 | 0,5 | nie | 10 |
| Nagłówek 6 ms | 18 | 6 | 2,03 | 0,7 | nie | 10 |
| Nagłówek 7 ms | 18 | 6 | 2,29 | 0,9 | nie | 10 |
| Nagłówek 8 ms | 18 | 6 | 2,54 | 1,1 | nie | 10 |

* + 1. Jak projektować style?

Jest sporo zasad typograficznych, które ogólnie dotyczą projektowania styli. Zasady przyjęte dla styli szablonów:

1. Treść jest przeznaczona przede wszystkim do czytania na ekranie urządzeń elektronicznych. W konsekwencji tekst nie jest justowany do obu krawędzi tekstu na stronie, lecz jest wyrównany do lewej krawędzi tekstu na stronie.
2. Treść jest możliwie gęsto upakowana na stronie. To wymaganie pochodzi z dawnych czasów. Celem było wyróżnienie / odróżnienie naszej dokumentacji od konkurencji.
3. Minimalne marginesy, by zwiększyć upakowanie treści na stronie.

Microsoft Word wymusza dodatkowe, specyficzne zasady tworzenia styli:

* 1. Jak najmniej pustych paragrafów (¶).
  2. Tabele są umieszczone jakby poza warstwą tekstu, tzn. że paragraf, w którym umieszczona jest tabela, ma zerowy odstęp od góry i zerowy odstęp od dołu.

Podobnie specyficzne zasady wymusza tworzenie dokumentów technicznych. Itd. Itp.

Zasady, które przyjęto:

(…)

* 1. Makra (Macros\_ms.dotm)
  2. Przygotować czysty plik Normal.dotm.
     1. Ustalić, na jaki folder wskazuje ustawienie konfiguracyjne Microsoft Word User templates.   
        Domyślnie jest to folder c:\Users\<NazwaUzytkownika>\AppData\Roaming\Microsoft\Templates\.
     2. Przejść do tego folderu.
     3. Wyłączyć Microsoft Word.  
        Wszystkie instancje.
     4. Zmienić nazwę pliku: Normal.dotm → Normal.old.   
        Ma to zabezpieczyć plik przed usunięciem.
     5. Uruchomić Microsoft Word.  
        Sprawdzić, czy w folderze User templates powstał nowy plik Normal.dotm.  
        Jeżeli nie jest widoczny, odświeżyć zawartość folderu. Jeżeli nadal nie jest widoczny, zamknąć Microsoft Word i sprawdzić ponownie.
  3. Ustalić, na jaki folder wskazuje ustawienie konfiguracyjne Microsoft Word Startup.  
     Domyślnie jest to folder C:\Users\<NazwaUzytkownika>\AppData\Roaming\Microsoft\Word\startup\.
  4. Skopiować plik Normal.dotm do folderu Startup.
  5. Zamknąć Microsoft Word.
  6. Usunąć plik Normal.dotm z folderu User templates.
  7. Zmienić nazwę pliku Normal.old → Normal.dotm.
  8. W folderze Startup zmienić Nazwę pliku Normal.dotm → Macros\_ms.dotm.
  9. Otworzyć plik Macros\_ms.dotm. Otworzyć menadżer kodu VBA (Alt + F11).
  10. Zaimportować moduły z kodem z repozytorium <https://svn.tens.pl/CompanyTemplates/MakraWord/20250217_ms>.
      1. Jak sprawdzić, czy kod makr daje się skompilować

Microsoft Visual Basic for Applications – Macros\_ms > Debug > Compile TemplateProject

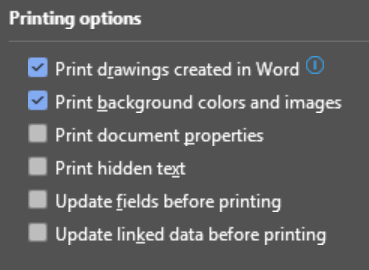
To polecenie podejmuje próbę kompilacji całego kodu, wszystkich modułów.



Nie da się zaimportować treści do modułu Microsoft Word Objects → ThisDocument z pliku ThisDocument.cls. Trzeba w tym celu napisać makro.

Nie da się go też ręcznie usunąć.

* + 1. Printing options



Pic. 40. Example. [source: ]

Word Option Name VBA Property Name

Print drawings created in Word Options.PrintDrawingObjects

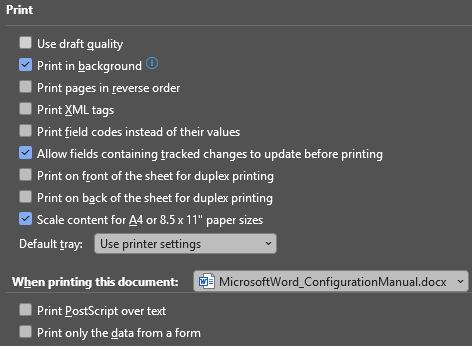
Print background colors and images Options.PrintBackgrounds

Print document properties Options.PrintProperties

Print hidden text Options.PrintHiddenText

Update fields before printing Options.UpdateFieldsAtPrint

Update linked data before printing Options.UpdateLinksAtPrint



Pic. 41. Example. [source: ]

.PrintDraft = False ' Use draft quality

.PrintBackground = True ' Print in background

.PrintReverse = False ' Print pages in reverse order

.PrintXMLTag = False ' Print XML tags

.PrintFieldCodes = False ' Print field codes instead of their values

' Allow fields containing tracked changes to update before printing, not available in VBA

.PrintOddPagesInAscendingOrder = False ' Print on front of the sheet for duplex printing

.PrintEvenPagesInAscendingOrder = False ' Print on back of the sheet for duplex printing

' Scale content for A4 or 8.5 x 11'' paper size, not available in VBA

.DefaultTray = wdPrinterDefaultBin ' Default tray: User printer settings

* 1. Fonts
     1. Where fonts file are located?

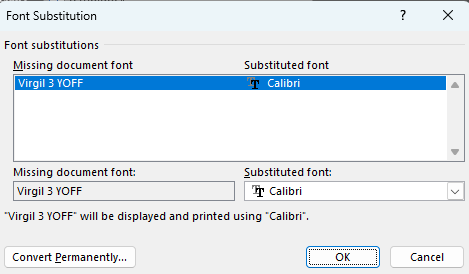
System fonts: C:\Windows\Fonts.

Fonts installed by user: c:\Users\<YourUserName>\AppData\Local\Microsoft\Windows\Fonts\.

* + 1. How do you check if some fonts aren’t missing in my document?

Answering this question can be handy in case you used specific fonts in your document and e.g. on purpose didn’t embed them within your DOCX document.

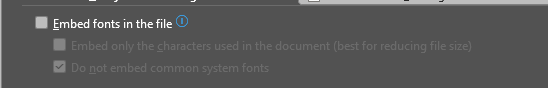
File > Options > Advanced > Show document content > Font Substitution… button.



Pic. 42. Font substitution. [source: herein]

* + 1. How to embed fonts within your document?

File > Options > Save > Embed fonts in the file.



Pic. 43. Embed fonts in the file. [source: herein]

* + 1. Default font

The default font are dedicated Excalifont-Regular.ttf fonts:

* 1. for document body is Excalifont-Regular.ttf aka +Body, following naming convention of Microsoft Word,
  2. for document headers is Excalifont-Regular.ttf is aka +Headings, following naming convention of Microsoft Word.

For meaning of Body and Headings see chapter 6.4.

* + 1. Features of Excalifont-Regular.ttf fonts
  1. Not a monospace, but proportional.
  2. Sans-serif.
  3. Prone to [homoglyphs issue](https://en.wikipedia.org/wiki/Homoglyph).
     1. Font math

Domyślny rozmiar: 11 pkt. To daje odpowiednio 11 × 0.03515 mm = 0,38665 mm. Biorąc pod uwagę wysokość strony przy naszych marginesach (wysokość użyteczna strony to 26,3 cm) oraz pojedynczą interlinię (120% wysokości czcionki), to daje ≈ 67 wierszy tekstu.

Dopuszcza się w wyjątkowych sytuacjach zmianę rozmiaru na 10 pkt. (schematy blokowe) czy nawet 8 pkt. (nagłówki i stopki).

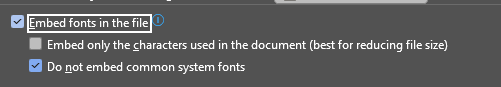
Dla czcionki Calibri wychodzi średnio 14 słów na linijkę (1 słowo = 5 znaków + spacja = 6 znaków). Szerokość znaku to średnio połowa wysokości (5,5 punktu).

Ze względu na powyższe cechy do zastosowań technicznych, np. prezentacji kodu źródłowego, konieczne jest wykorzystanie jeszcze jednej czcionki. Wybór padł na Consolas.

In case document content to be edited by external company, it is reasonable to embed in it fonts to keep the same look and feel.

| **⚠** Warning |
| --- |
| The font files can follow specific licensing and may be copyrighted. |

Embedding fonts is seldomly practiced and because of that it is advisable to embed fonts manually using GUI of Microsoft Word. On time when a file where we want to embed fonts is open Enter Microsoft Word menu File > Options > Save and check ☑ the options as in Pic. 20. Save the file.



Pic. 44. Example. [source: herein]

Please note that its size can be significantly icnreased. For example:

* + Wielkość niemal pustego pliku przed osadzeniem czcionki: fikumiku.docx 14 048 B.
  + Wielkość tego samego pliku po osadzeniu czcionki: fikumiku.docx 352 981 B
    1. Skąd pobrać czcionki Excalifont-Regular.ttf?

(…)

* + 1. Jak wygląda rodzina czcionek Excalifont-Regular.ttf?

Excalifont-Regular.ttf To jest przykład.

* + 1. Czy proces osadzania czcionek jest odwracalny?

Czyli czy można usunąć wcześniej osadzone czcionki z tego pliku. Tak, proces jest odwracalny. Wystarczy wykonać te same kroki, ale tym razem odznaczyć opcję pokazaną na Pic. 3.

* + 1. Fonts management

There is no way to list embedded fonts by VBA in Microsoft Word.

To check which fonts are actually embedded in your Microsoft Word document, you can follow these steps:

1. Save your Word document.

2. Change the file extension from .docx to .zip.

3. Extract the contents of the .zip file.

4. Open the extracted folder and navigate to the word folder.

5. Look for a file named fontTable.xml and open it with a text editor.

6. In the fontTable.xml file, you will see a list of all the fonts used in the document. Embedded fonts will have an attribute indicating that they are embedded.

' To recognize which fonts are embedded in your document by looking at the fontTable.xml file, you need to look for the <w:embedRegular>, <w:embedBold>, <w:embedItalic>, and similar elements within each <w:font> element. These elements indicate that the font is embedded.

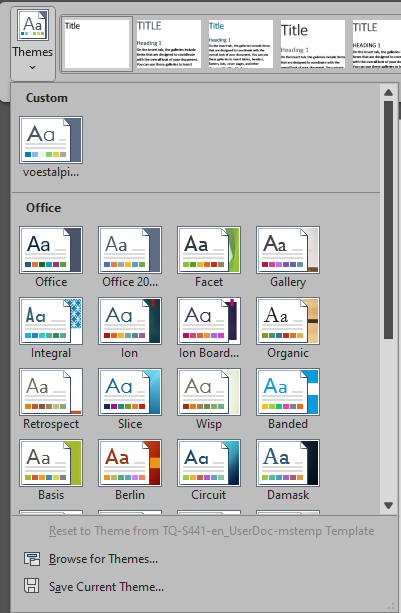
* 1. Theme (Theme\_ms.thmx)

Motyw to jedna z domyślnych cech stosowanych przez procesor tekstu Microsoft Word dla treści (oprócz szablonu i stylu). Innymi słowy bez względu na to, czy robimy to świadomie, czy nie, wzystkie nasze dokumenty wykorzystują motyw.

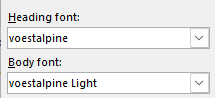
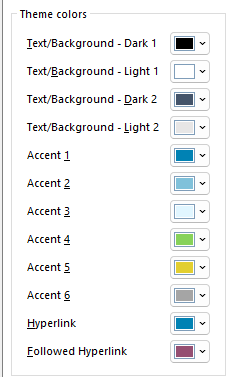
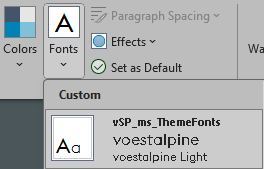
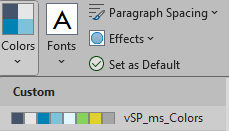
Zaprojektowanie i wykorzystanie własnego motywu pozwala przede wszystkim na szybkie i wygodne wykorzystanie domyślnych kolorów korporacyjnych.

Zdefiniowany plik motywu: Theme\_ms.thmx.

By zobaczyć wśród zdefiniowanych motywów także te zdefiniowane przez nas, pliki muszą być umieszczone w folderze   
C:\Users\<UserName>\AppData\Roaming\Microsoft\Templates\Document Themes



Pic. 45. Menu Design > Themes, customized themes at the top. [source: herein]



Pic. 46. Theme settings. [source: herein]

* 1. Keyboard shortcuts

Keyboard shortcuts, especially connected with the possibility to apply styles, make Microsoft Word still unparalleled tool in terms of speed content edition. In this chapter I did my best to convince you to apply shortcuts in your everyday practice. Generally working with mouse is one of the most important factor in terms of efficiency: those of us who most of the time work with mouses are much less efficient.

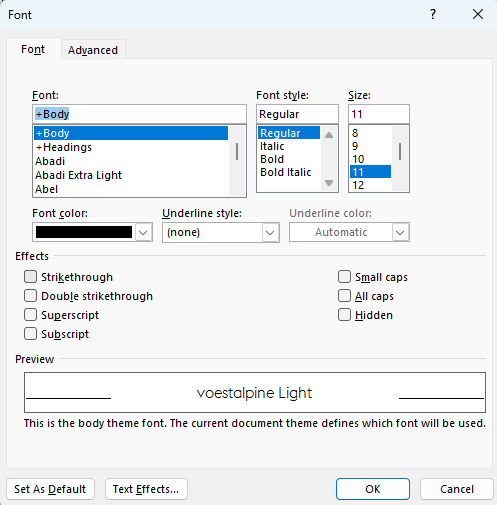
* + 1. What are the keyboard shortcuts?

Computer keyboard is used most of the time to type characters serially. For comparison a music keyboard is used to press its keys in parallel. This principle stays behind a concept of music chords or sound combinations. Combination of specific sounds make listening music more enjoyable contrary to serial sounds.

Also computer keyboard let us to enter characters in parallel. By analogue also concurrently pressed characters sometimes are referred as chords, but most of the time as (keyboard) shortcuts. You can imagine that pressing of a shortcut could produce a pleasant sound.

* + 1. Why keyboard shortcuts in Microsoft Word?

Microsoft Word could be seen as a huge container of many functions. Only small fractions of those functions are available by default in the GUI. It is true that GUI is configurable to some degree, but still finding and applying some functions with mouse navigation is time consuming. The alternative approach is to define keyboard shortcuts at least for some most time consuming functions. The idea of shortcuts is based on reusability of existing keys and combining them to call specific functions.



Pic. 47. Text strikethrough. [source: herein]

My fevorite example is striking through text. Personally, I find it very useful to strike through edited content, without removing it. It mimics natural edition habits acquired on time of working with pen and paper. How many time one have to click with a mouse to strike through a specific word in a sentence?

Let's test:

1. Rich with a mouse specific word in a specific sentence.
2. Select it, by pressing down left mouse key and drag.
3. Click in the Microsoft Word menu at Home.
4. Click in a Font group bottom right corner of ribbon section. It is hard to click such a small icon, isn't it?
5. Click the Strikethrough effect.
6. Click ok.

The same task, if accomplished with keyboard:

1. Rich specific word in a specific sentence (e.g. Ctrl + → / ↑ / ↓ / ←) × repeated.
2. Select it (e.g. Ctrl + Shift + →).
3. Call a toggle macro: Ctrl + Shift + X.

What would be easier and quicker?

* + 1. Is there an alternative approach?

There is an alternative approach. Imagine that each required function is assigned to a dedicated keyboard key. Usually desktop keyboard has around 100 keys. How many keys should then be reserved to Microsoft Word? In my personal case about 70. There is a good chance that your keyoffice board already has 12 keys ready to be assigned to special purposes. These are so called "f-keys", e.g. F1 … F12. But what could I do with the rest of 58 keys if not apply the shortcuts?

Let's compare our working desk to a plane cockpit. The idea "one key, one function" was met there. Pilot has all over her / his body access to keys, switches and other manipulators. Just to let her / him for immediate access to any desired function.

Can we organize work in a similar matter and add e.g. dedicated keyboard just to call keyboard shortcuts and macros? The answer is yes, definitely, but this is a subject for dedicated chapter, if not a book. You can do it yourself with relatively low cost and effort.

* + 1. What are advantages and disadvantages of keyboard shortcuts?

Tab. 7. Shortcuts advantages and disadvantages. [source: herein]

|  |  |  |
| --- | --- | --- |
| advantages | disadvantages | comment |
| easier access to specific functions | need to memorize shortcuts to really push the limits | dedicated print-outs or even keeping windows containing shortcuts on a screen |
| quicker access to specific functions | misses could run undesired functions |  |

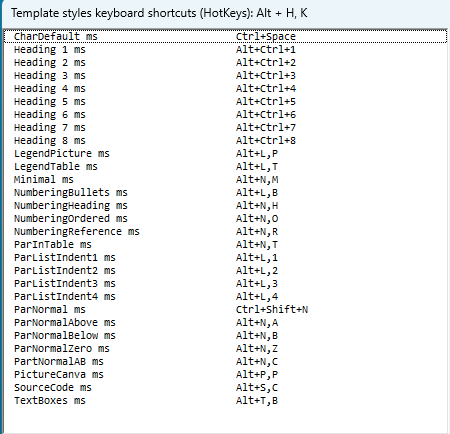
* + 1. What have I done to make your life easier?

Shortcuts are sometimes referred to as hotkeys. The wider, but related concept, is calling sequences of actions upon entering sequence of characters. Sequences of characters are often referred in IT as strings, so by analogue the full name of such a sequence would be a hotstring.

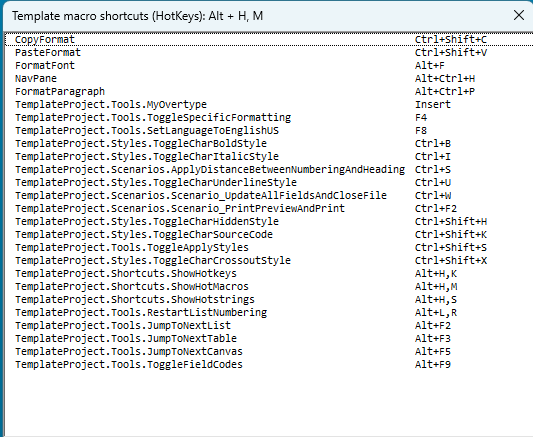
In Microsoft Word one can find both concepts, hotkeys and hotstrings. Hotstrings are named in Microsoft Word as autotext. Autotext can be combined with building blocks.

I decided to reinforce my memory with dedicated small windows, which show me keyboard shortcuts defined within my configuration files directly on my screen. Most of the time I work nowadays with at least 3× screens: laptop + 2× office size. Laptop screen is much smaller in size and sits a bit aside of two main screens. It makes almost perfect arrangement to keep windows with lists of keyboard shortcuts so close to my working place.

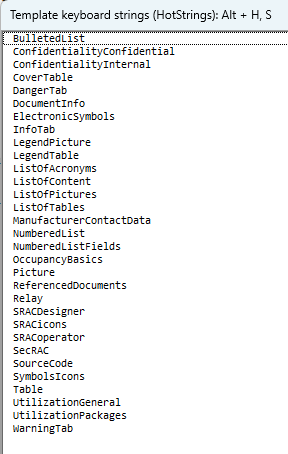
To call those windows I have to memorize just 3× keyboard shortcuts or… reopen any existing Microsoft Word file or open a new file. User is always asked if she/he wishes to open those windows on time of such events.



Pic. 48. Style hotkeys. [source: herein]



Pic. 49. Command hotkeys. [source: herein]



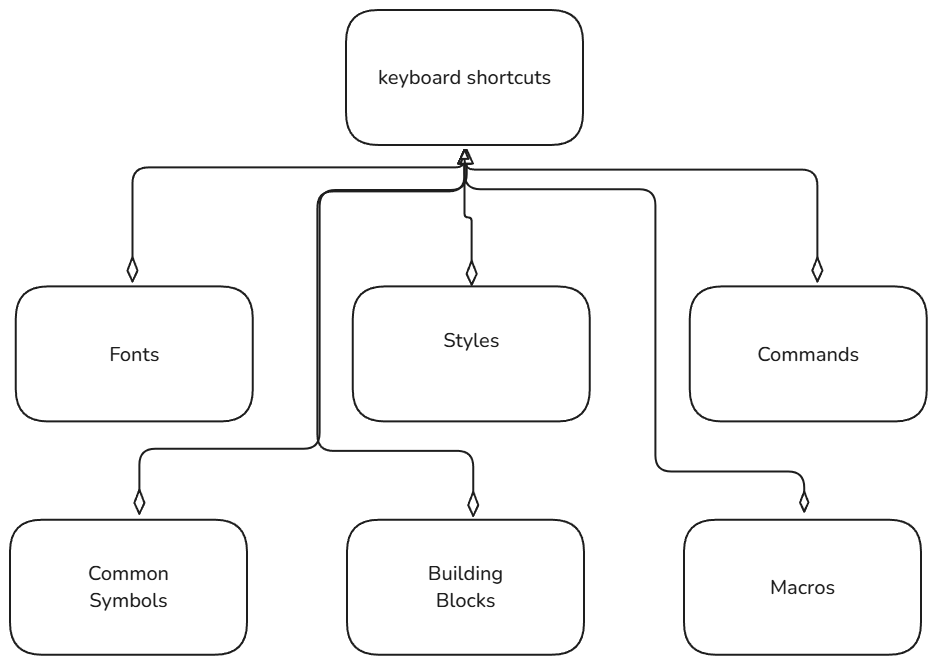
Pic. 50. Building blocks hotstrings. [source: herein]

* + 1. Keyboard shortcuts chaining

One of the useful tricks you can do in VBA I called a shortcuts chaining. For example I find it quite annoying that upon saving a DOCX file which contains Table of Contents, or any other fields, those are not automatically updated. I prepared a dedicated macro, which updates all the fields and then calls standard function which saves the content.

* + 1. Keyboard shortcuts classification in Microsoft Word

Internally Microsoft Word classifies keyboard shortcuts as in Pic. 51. Additionally it enables to save keyboard shortcuts according to context of your choice.



Pic. 51. Keyboard shortcuts, types classification. [source: MicrosoftWordTemplateConcept.excalidraw]

* + 1. Keyboard shortcuts and context
    2. Listing of all customized shortcuts and relationship to macros

Macro Name: CountAndShowAllShortcuts

Date: 2025-07-16

Time: 22:46:34

Template Name: C:\Users\v523580\AppData\Roaming\Microsoft\Templates\Styles\_ms.dotm

Total Shortcuts: 55

Shortcuts category AutoText:

Shortcuts category Command:

1. CopyFormat - Ctrl+Shift+C

2. PasteFormat - Ctrl+Shift+V

3. FormatFont - Alt+F

4. NavPane - Alt+Ctrl+H

5. FormatParagraph - Alt+Ctrl+P

Shorcuts category Disable:

Shortcuts category Font:

Shortcuts category Macro:

1. TemplateProject.Tools.MyOvertype - Insert

2. TemplateProject.Tools.ToggleSpecificFormatting - F4

3. TemplateProject.Tools.SetLanguageToEnglishUS - F8

4. TemplateProject.Styles.ToggleCharBoldStyle - Ctrl+B

5. TemplateProject.Styles.ToggleCharItalicStyle - Ctrl+I

6. TemplateProject.Scenarios.ApplyDistanceBetweenNumberingAndHeading - Ctrl+S

7. TemplateProject.Styles.ToggleCharUnderlineStyle - Ctrl+U

8. TemplateProject.Scenarios.Scenario\_UpdateAllFieldsAndCloseFile - Ctrl+W

9. TemplateProject.Scenarios.Scenario\_PrintPreviewAndPrint - Ctrl+F2

10. TemplateProject.Styles.ToggleCharHiddenStyle - Ctrl+Shift+H

11. TemplateProject.Styles.ToggleCharSourceCode - Ctrl+Shift+K

12. TemplateProject.Tools.ToggleApplyStyles - Ctrl+Shift+S

13. TemplateProject.Styles.ToggleCharCrossoutStyle - Ctrl+Shift+X

14. TemplateProject.Shortcuts.ShowHotkeys - Alt+H,K

15. TemplateProject.Shortcuts.ShowHotMacros - Alt+H,M

16. TemplateProject.Shortcuts.ShowHotstrings - Alt+H,S

17. TemplateProject.Tools.RestartListNumbering - Alt+L,R

18. TemplateProject.Tools.JumpToNextList - Alt+F2

19. TemplateProject.Tools.JumpToNextTable - Alt+F3

20. TemplateProject.Tools.JumpToNextCanvas - Alt+F5

21. TemplateProject.Tools.ToggleFieldCodes - Alt+F9

Shortcuts category Nil:

Shortcuts category Prefix:

Shortcuts category Style:

1. CharDefault ms - Ctrl+Space

2. ParNormal ms - Ctrl+Shift+N

3. ParListIndent1 ms - Alt+L,1

4. ParListIndent2 ms - Alt+L,2

5. ParListIndent3 ms - Alt+L,3

6. ParListIndent4 ms - Alt+L,4

7. NumberingBullets ms - Alt+L,B

8. LegendPicture ms - Alt+L,P

9. LegendTable ms - Alt+L,T

10. ParNormalAbove ms - Alt+N,A

11. ParNormalBelow ms - Alt+N,B

12. PartNormalAB ms - Alt+N,C

13. NumberingHeading ms - Alt+N,H

14. Minimal ms - Alt+N,M

15. NumberingOrdered ms - Alt+N,O

16. NumberingReference ms - Alt+N,R

17. ParInTable ms - Alt+N,T

18. ParNormalZero ms - Alt+N,Z

19. PictureCanva ms - Alt+P,P

20. SourceCode ms - Alt+S,C

21. TextBoxes ms - Alt+T,B

22. Heading 1 ms - Alt+Ctrl+1

23. Heading 2 ms - Alt+Ctrl+2

24. Heading 3 ms - Alt+Ctrl+3

25. Heading 4 ms - Alt+Ctrl+4

26. Heading 5 ms - Alt+Ctrl+5

27. Heading 6 ms - Alt+Ctrl+6

28. Heading 7 ms - Alt+Ctrl+7

29. Heading 8 ms - Alt+Ctrl+8

Shortcuts category Symbol:

* + 1. Copying of all customized shortcuts to TXT files

Content of every MsgBox in Microsoft Windows can be just copied with keyboard shortcut Ctrl + C to system memory named clipboard. Unfortunately this is not the case with content of the GUI window or form window. Because of this drawback the following macro was prepared: ShortcutsToTxt. The output files are created in wdDocumentsPath folder.

* + 1. List of all commands and related keyboard shortcuts

See Appendix A MWAllCommandsList.docx and Appendix B MWKeyboardShortcutsList.docx .

1. Jak wygląda relacja szablon DOTM ↔ plik DOCX

Po dołączeniu niestandardowego szablonu do pliku DOCX w programie Microsoft Word, za kulisami dzieje się kilka rzeczy związanych ze stylami i sposobem zarządzania nimi w dokumencie. Ważne jest jednak, aby dokładnie wyjaśnić, jak działa ten proces i co to oznacza dla relacji pomiędzy plikiem szablonu (DOTM) a plikiem DOCX.

* 1. Co się dzieje po dołączeniu do pliku DOCX pliku szablonu (DOTM)?

**Importowanie stylów**: Po dołączeniu szablonu do dokumentu programu Word style zdefiniowane w szablonie stają się dostępne w dokumencie. Oznacza to, że wszystkie unikalne style w szablonie, które jeszcze nie istnieją w pliku DOCX, są dodawane do listy stylów dokumentu.

**Zastępowanie stylów**: Jeśli w dokumencie znajdują się style o takich samych nazwach jak style w szablonie, dołączenie szablonu może zmodyfikować te style w dokumencie, aby odpowiadały definicjom szablonu, w zależności od ustawień programu Word i sposobu zastosowania szablonu.

**Brak ciągłego łącza:** W przeciwieństwie do niektórych innych programów opartych na szablonach, Word nie utrzymuje dynamicznego łącza do szablonu w celu definiowania stylów. Po zaimportowaniu lub zaktualizowaniu stylów w pliku DOCX, gdy szablon jest dołączony, style te są osadzone w samym pliku DOCX.

* 1. Co dzieje się po odłączeniu lub niedostępności szablonu?

**Zachowanie stylu**: Po zaimportowaniu stylu z szablonu do dokumentu programu Word, definicja stylu zostaje osadzona w dokumencie. Oznacza to, że nawet po odłączeniu szablonu lub przeniesieniu pliku DOCX na inny komputer, na którym szablon nie jest dostępny, style z szablonu, które zostały zastosowane lub dodane do dokumentu, pozostają nienaruszone.

**Brak zależności**: Dokument nie zależy od szablonu, aby zachować definicje stylów. Po zaimportowaniu lub zmodyfikowaniu stylów w dokumencie za pośrednictwem szablonu, stają się one częścią samego dokumentu. Style te można edytować, usuwać lub modyfikować niezależnie od oryginalnego szablonu.

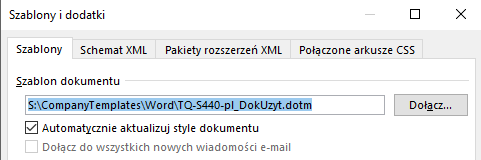
* 1. Prawda czy fałsz: Definicje stylów są przechowywane w pliku DOCX

**Prawdziwe stwierdzenie**: Tak, prawdą jest, że po zastosowaniu lub dodaniu definicji stylów niestandardowego pliku szablonu do pliku DOCX, definicje te są przechowywane w samym pliku DOCX. Oznacza to, że dokument zachowuje swoje formatowanie i style, nawet jeśli oryginalny szablon nie jest już dostępny lub dołączony.

Podsumowując, dołączenie niestandardowego szablonu w programie Word jest sposobem na szybkie zastosowanie zestawu stylów i standardów formatowania do dokumentu, ale po zastosowaniu tych stylów stają się one częścią dokumentu i nie wymagają, aby szablon pozostawał połączony lub obecny.

* 1. Czy style zawsze są poprawnie przenoszone z szablonu do pliku?

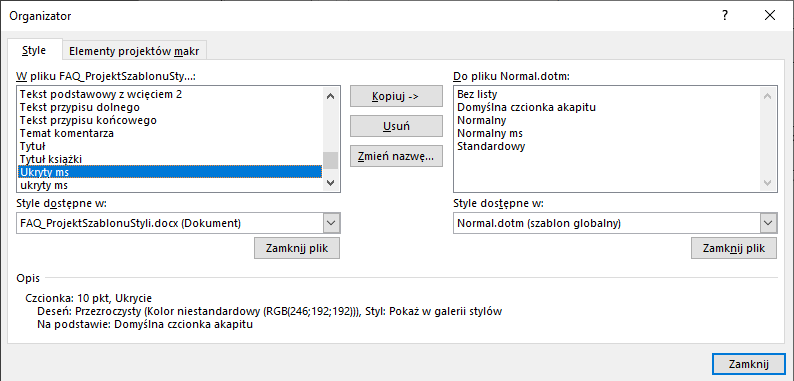
Niestety nie. Zdarza się, że w pliku DOCX już jest jakiś styl. Jego definicja jest inna w szablonie. Chcemy, by nowa definicja nadpisała starą. Wymuszamy to poprzez ustawienie *Automatycznie aktualizuj style dokumentu*, ale to nie nic nie daje. Nadal stara definicja stylu nie ulega zmianie. Niestety nie ma żadnej reguły, która pozwoliłaby przewidzieć, kiedy tak się stanie.



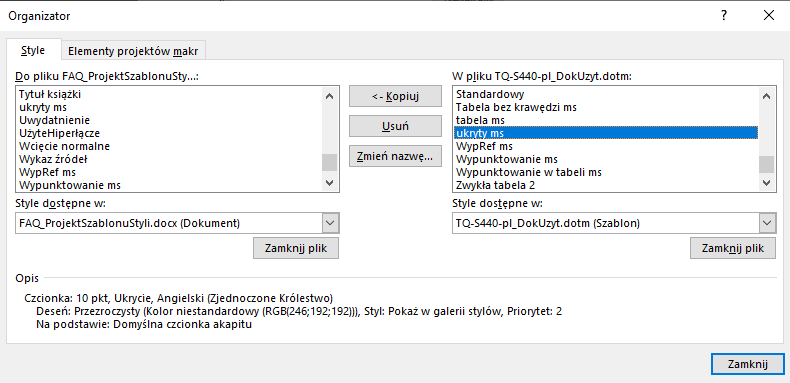
Rys. 2. Automatycznie aktualizuj style dokumentu.

* 1. Co zrobić, jeżeli definicja stylu nie zostaje przeniesiona do pliku z szablonu?

W takiej sytuacji należy ręcznie usunąć styl z pliku DOCX i następnie ręcznie go do tego pliku DOCX przenieść z szablonu DOTM. Na pytanie z okna dialogowego odpowiadamy *Tak na wszystkie.*



Rys. 3. Przykład, usuwanie stylu *Ukryty ms* z pliku FAQ.



Rys. 4. Przykład, dodawanie stylu *ukryty ms* z pliku szablonu.

Uwaga. Usunięcie stylu z pliku DOCX oznacza usunięcie formatowania we wszystkich miejscach w dokumencie. W konsekwencji po przeniesieniu stylu z pliku DOTM do pliku DOCX, w pliku DOCX trzeba ponownie odwiedzić wszystkie miejsca, gdzie był przypisany stary styl i je zaktualizować.

1. Pliki i prawo

Przez pliki rozumiemy tak plik szablonu (DOTM) jak i pliki powstające na bazie szablonu (DOCX).

* 1. Pliki dokumentów (DOCX) i prawo

W tym rozdziale omówiono zagadnienia prawne dotyczące dokumentów (plików DOCX) powstających na bazie szablonów (plików DOTM).

* + 1. W jaki sposób chronimy treść w dokumentach na bazie szablonów?

Za pomocą bloku konstrukcyjnego *Przeznaczenie* z notą prawną wyjaśniającą sposób ochrony własności intelektualnej. Ten blok konstrukcyjny jest składową bloku konstrukcyjnego *Druga strona*. Jest umieszczany u góry strony, nad tabelą autor / sprawdzający / zatwierdzający. Jeżeli korzystasz z plików szablonów, to już się tam znajduje.

Za pomocą klasyfikacji poufności treści umieszczoną na każdej stronie, w stopce dokumentu.



Fig. 1. Przykład klasy poufności *Dokument wewnętrzny*.

Wyboru klasy poufności dla dokumentu dokonujemy za pomocą makra *Wypełnienie pierwszej strony*. Dostępne klasy poufności, na podstawie dokumentu TQ-I501.

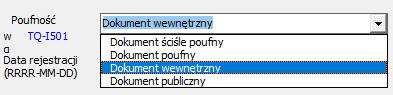


Fig. 2. Lista rozwijana, poufność.

* + 1. Jaką klasę poufności wybrać?

Regulamin [*Bezpieczeństwo informacji*](https://team.voestalpine.net/site/6410/ProceduryInstrukcjeRegulaminy/Bezpiecze%C5%84stwo%20Informacji.pdf) (rozdz. 4 → rys. 1) nie udziela jasnej odpowiedzi na to pytanie. Wobec tego przygotowano (MS @ 2024-03-12) poniższe zestawienie.

Tab. 2. Klasa poufności a kategoria dokumentu.

|  |  |  |
| --- | --- | --- |
| kategoria poufności | kategoria dokumentu | przykłady |
| dokument publiczny  public | ulotka, informacje na stronie WWW | ogłoszenie o pracę, kanały LinkedIn |
| dokument wewnętrzny  internal document | dokumentacja użytkownika | DTR / System Manual, Plan Utrzymania / Warunki Utrzymania / Operation & Maintenance Manual, Podręcznik SRAC / SRAC User Manual |
| dokument poufny  confidential | dokumentacja techniczna | rysunki projektowe, dokumentacja kodu źródłowego, kod źródłowy, koncepcje  pełne wydania raportów z badań |
| dokument ściśle poufny  strictly confidential | dokumenty finansowe, dokumenty z informacjami wrażliwymi | raporty wyjaśniające szczegóły konstrukcyjne, przeznaczone dla ściśle określonego odbiorcy |

* + 1. Nadal nie wiem, jaką klasę poufności wybrać, co mam zrobić?

Spytać przełożonego, ewentualnie skontaktować się z osobą odpowiedzialną za opiekę prawną. Dane osobowe tej osoby są dostępne w Sekretariacie.

* + 1. W jakiej formie wysłać dokument do klienta?

W zasadzie zawsze powinien być to plik PDF (tylko do odczytu, z ograniczoną możliwością edycji). Jeżeli ma być to plik również w postaci źródłowej (np. DOCX) należy zadbać o to, by był on prawidłowo wyświetlony na komputerze partnera biznesowego, a więc zapewnić mu plik szablonu.

* + 1. W jaki sposób wyjaśnić klientowi oznaczenie klasy poufności?

Blok konstrukcyjny: ConfidentialityInternal, ConfidentialityConfidential.

Odpowiedni blok konstrukcyjny, w zależności od wybranej klasy poufności, należy umieścić ręcznie w nowych plikach, w rozdziale *Introduction* jako podrozdział pt. *Confidentiality of the document*.

[Źródło: Poniższa definicja bloków konstrukcyjnych została przygotowana na postawie e-maila od opiekuna prawnego z 2024-03-07 14:36. Edycja: MS].

Internal document: This means that the content of this document is intended for individual business partners, based on individual agreements, e.g. non-disclosure agreements (NDA) or provisions between business partners (B2B). Not for public use without written permission. The English (-en) version of this document is the original version. All non-English editions of this document are translations of the original. This document is intended for the exclusive use of persons operating the system. Without written consent, this document must not be made available to third parties.

Dokument wewnętrzny: Treść tego dokumentu jest przeznaczona dla indywidualnych partnerów biznesowych, na podstawie indywidualnych umów, np. umowy o zachowaniu poufności (NDA) lub innej umowy zawartej pomiędzy partnerami biznesowymi (B2B). Nie jest przeznaczony do użytku publicznego bez pisemnej zgody. Wersja angielska (-en) tego dokumentu jest wersją oryginalną. Wszystkie wydania tego dokumentu w języku innym niż angielski są tłumaczeniami oryginału. Niniejszy dokument jest przeznaczony wyłącznie do użytku przez przeznaczony do tego personel. Bez pisemnej zgody niniejszy dokument nie może być udostępniany osobom trzecim.

Confidential document: This means that the content of this document it is dedicated only to business partners, based on individual agreements, e.g. Non-Disclosure Agreement (NDA) or provisions between business partners (B2B). Not for public use without written permission and it should be specially protected against any unauthorized persons, in particular: copies should be left in locked places, mailing with this information should be encrypted and the list of persons with access to this information should be known. The English (-en) version of this document is the original one. All non-English language editions of this document are translations of the original one. This document is intended for the exclusive use of persons operating the system. Without written agreement, this document must not be made available to third parties.

Dokument poufny: Treść tego dokumentu jest przeznaczona dla indywidualnych partnerów biznesowych, na podstawie indywidualnych umów, np. umowy o zachowaniu poufności (NDA) lub innej umowy zawartej pomiędzy partnerami biznesowymi (B2B). Nie jest przeznaczony do użytku publicznego bez pisemnej zgody i należy go szczególnie chronić przed osobami nieuprawnionymi, w szczególności: kopie należy pozostawiać w zamkniętych miejscach, wiadomości przesyłane za pomocą poczty elektronicznej z tą informacją powinny być szyfrowane oraz powinna być zdefiniowana lista osób mających dostęp do tych informacji. Wersja angielska (-en) tego dokumentu jest wersją oryginalną. Wszystkie wydania tego dokumentu w języku innym niż angielski są tłumaczeniami oryginału. Niniejszy dokument jest przeznaczony wyłącznie do użytku przez przeznaczony do tego personel. Bez pisemnej zgody niniejszy dokument nie może być udostępniany osobom trzecim.

* + 1. Nie jestem pewna / pewny, czy mogę wysłać dany dokument. Co mam zrobić?

Spytać partnera biznesowego, czy podpisał umowę o poufności (ang. NDA, Non-Disclosure Agreement). Poinformować o zasadach poufności, ochronie praw autorskich. Jeżeli te warunki są spełnione, wysłać dokument w postaci PDF.

Jeżeli warunki z punktu a nie są spełnione, spytać przełożonego.

* 1. Pliki szablonów (DOTM) i prawo

W tym rozdziale omówiono zagadnienia prawne dotyczące wykorzystania szablonów (plików DOTM).

* + 1. Czy szablony są objęte prawem autorskim?

(…) fikumiku

* + 1. Czy wysyłając do klienta plik szablonu wysyłam mu jednocześnie czcionki?

Similar to DOCX files, when you send a Microsoft Word template file (.DOTM), it does not include the fonts used to author the file. The .DOTM file will reference the fonts, but it relies on those fonts being installed on the recipient's computer to display the document as intended. If the recipient doesn't have the necessary fonts, Word will use a default or alternative font, which may change the appearance of the document.

However, you have the option to embed fonts within the template:

1. Open the Word template (.DOTM) and go to File > Options > Save.
2. Check the Embed fonts in the file option under Preserve fidelity when sharing this document.  
   You can also choose to embed only the characters used in the template to reduce the file size or embed all characters for full editing capabilities.
3. Save the template.

When you embed fonts in your template, anyone opening the .DOTM file will see the document with the correct fonts, even if they don't have those fonts installed. This approach is especially useful when the template uses unique or less common fonts.

Remember, embedding fonts can significantly increase the file size, and you should ensure you have the legal right to embed and distribute any fonts used in your documents, as this can be restricted by the font's license.

1. Design of configuration

Less is more.

1. Dlaczego pliki szablonów często się zmieniają?

Ponieważ w tym samym pliku trzymamy definicje (stan na 2024-03-04):

styli, a jest ich 36,

bloki konstrukcyjne, a jest ich 31,

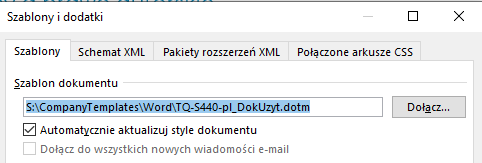
skróty klawiaturowe, a jest ich 14,

makra, a jest ich 29

To spora porcja danych konfiguracyjnych i kodu. W szczególności płynna jest liczba bloków konstrukcyjnych.

* 1. W jaki sposób zapewnić aktualizację wszystkich składników szablonu w konkretntym dokumencie?

W oknie Szablony i dodatki zaznaczyć Automatycznie aktualizuj style dokumentu. Wcisnąć przycisk OK.



Rys. 5. Aktualizacja składowych szablonu w dokumencie.

* 1. Kto jest odpowiedzialny za utrzymanie plików szablońow?

Zespół Academy & Training Center (ATC).

* 1. Czy pliki szablonów są nadzorowane przez system kontroli wersji?

Tak. Znajdują się w repozytorium SVN:

<https://svn.tens.pl/CompanyTemplates/OgolneZmakrami>

1. Ustawienia strony
   1. Marginesy
   2. Wielkości marginesów

Stosujemy marginesy lustrzane, wąskie. Celem jest upchanie jak największej liczby treści na stronie oraz umożliwienie bindowania wydrukowanych dokumentów.

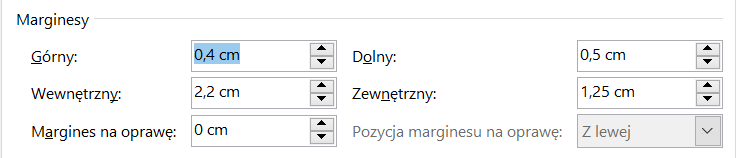


Fig. 3. Wielkości marginesów.

W konsekwencji przyjętych rozmiarów marginesów rozmiar strony dostępny do pokrycia przez tekst to:

szerokość: szerokość A4 (210 mm) − (2,2 cm + 1,25 cm) = 17,55 cm;

wysokość: wysokość A4 (297 mm) – (0,4 cm + 0,5 cm) = 28,8 cm.

nagłówek 1,25 cm, stopka 1,25 cm.

Szerokość i wysokość mają wpływ m.in. na maksymalną szerokość obrazków (kanwa) czy szerokość tabel.

* 1. Podział na wyrazy

Podział na wyrazy jest zapisany w szablonie. Domyślne ustawienie: podział automatyczny. Jest to świadome działanie mające na celu zmaksymalizowanie gęstości upakowania słów w linii tekstu.

1. Listy

Listy w Microsoft Word tworzy się z treści akapitów, nakładając na nią dwa typy styli, w podanej kolejności:

Style typu „Połączone (akapit i znak)”.

Style typu „Lista”.



Rys. 6. Przykładowa lista.

Na Rys. 6 mamy dwa akapity tekstu. Dla obu akapitów tekstu zastosowano dwukrotnie dwa typy styli:

Pierwszy rodzaj stylu to „Połączone (akapit i znak)” odpowiedzialne za tzw. wcięcie tekstu   
(kolor zielony; odległość tekstu w pionie od granicy tekstu na stronie).

Drugi rodzaj stylu to „Lista”   
(kolor fioletowy, w tym wypadku numerowanie za pomocą kolejnych liter alfabetu).

Na Rys. 6 widzimy wynik nadania obu akapitom obu styli. Zróbmy teraz to samo, ręcznie, krok po kroku.

Nazwy zastosowanych styli w powyższym przykładzie:

Styl typu „Połączone (akapit i znak)”: *ListaSeq 2 ms*.

Styl typu „Lista”: *Wypunktowanie ms*.

* 1. Pułapki

|  |
| --- |
| Ostrzeżenie |
| Domyślnie jawnie, w sposób półautomatyczny, Microsoft Word nadaje style typu „Połączone (akapit i znak)”, a style typu „Lista” nadaje automatycznie, często bez świadomości użytkownika, w sposób przypadkowy. |

Samo naciśnięcie znaków odpowiadających wypunktowaniom, np. sekwencji „1.·”, „a.·”, gdzie znak „·” (kropka) oznacza spację, powoduje dla nowych akapitów automatyczne nadanie styli tym akapitom jak dla list. Niestety te decyzje Microsoft Word podejmuje samowolnie. Jeżeli w aktualnie otwartym pliku mamy zdefiniowanych kilka styli typu „Lista” lub/i kilka styli typu „Połączone (akapit i znak)” dla list, to możemy tylko zgadywać, które style faktycznie zostaną przypisane automatycznie. Często dzieje się to niezgodnie z intencjami osoby wprowadzającej treść.

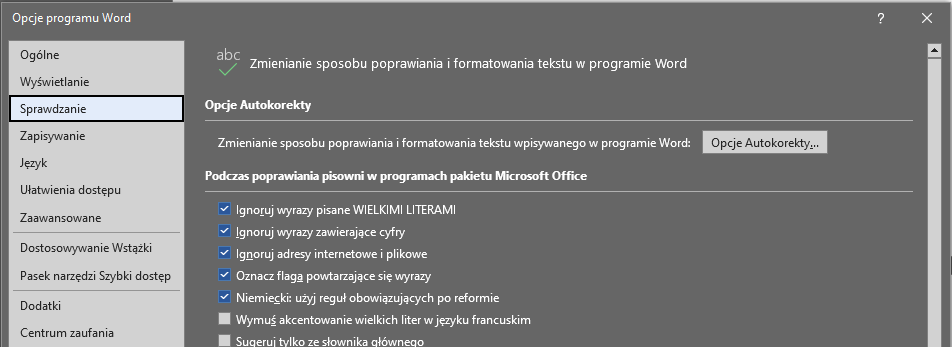
Przykładowo nigdy nie możemy być pewni, jaki styl typu lista zostanie nadany kolejnym akapitom tekstu oraz, w przypadku list numerowanych, jaka wartość zostanie nadana jako punktor. Co szczególnie dokuczliwe, błędy w numerowaniu list ujawniają się szczególnie dla dużych / długich dokumentów.

* + 1. Jak sobie z tymi pułapkami poradzić?

Istnieją co najmniej następujące obejścia:

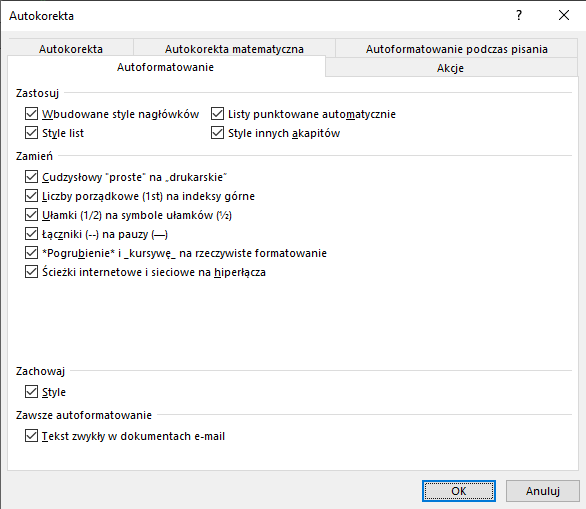
* 1. Nie ufać mechanizmowi automatycznego przypisywania styli do list, a nawet wprost go wyłączyć, patrz kolejny rozdział (metoda zalecana).
  2. Ręcznie nadawać style dla list (dla każdej listy dwukrotnie, tj. najpierw styl „akapit i znak”, a potem „lista”), patrz rozdział 16.2 (metoda zalecana).
  3. Ręcznie tworzyć numeracje list, w oparciu o mechanizm pól, patrz rozdział 16.4 (ostateczność, ze względu na czasochłonność).
     1. Wyłączenie automatycznej numeracji list

Plik → Opcje → Sprawdzanie



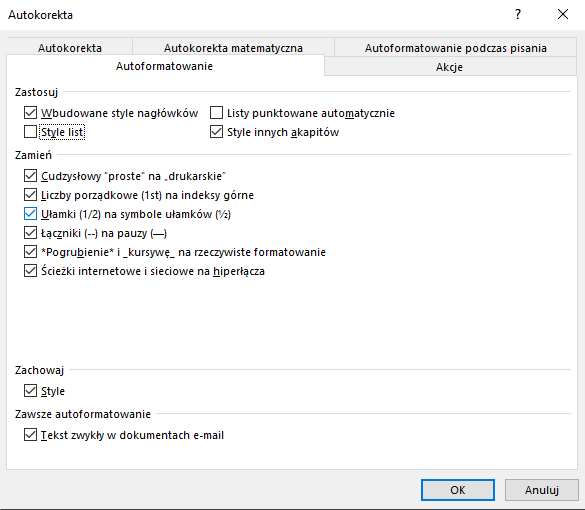
Rys. 7. Zakładka „Sprawdzanie”.

Przycisk „Opcje Autokorekty…” → zakładka „Autoformatowanie” (Rys. 8).



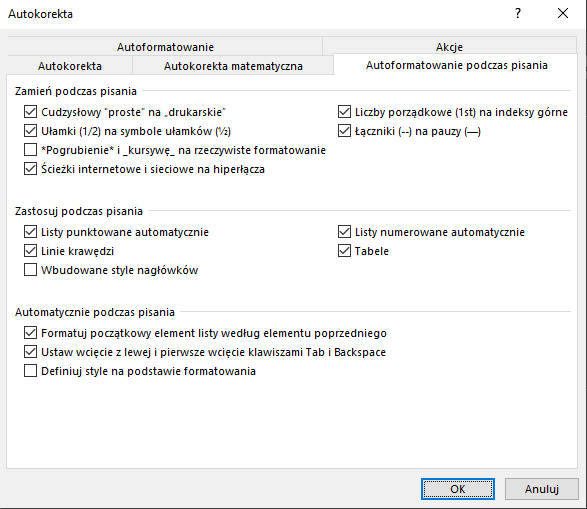
Rys. 8. Zakładka „Autoformatowanie”.

Zalecam pozostawienie ustawień, jak poniżej, patrz Rys. 9.



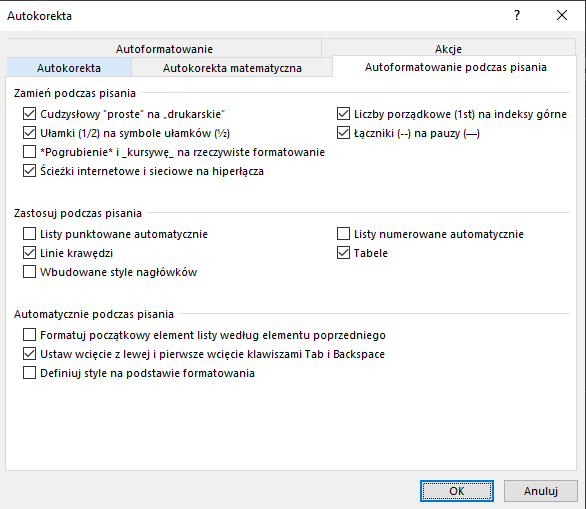
Rys. 9. Autokorekta.

Oraz dla zakładki „Autoformatowanie podczas pisania” zmianę ustawień domyślnych (Rys. 8).



Rys. 10. Autoformatowanie podczas pisania.

Na następujące (patrz Rys. 11).



Rys. 11. Autoformatowanie podczas pisania.

* 1. Jak ręcznie nadawać style dla list?
     1. Krok 0: akapity tekstu z domyślnym stylem akapitu



granica tekstu na stronie

margines strony

Rys. 12. Po nadaniu domyślnego stylu dla obu akapitów.

Na Rys. 12 widoczne są dwa przykładowe akapity treści, którym nadano domyślny styl akapitu *Normalny ms;* skrót klawiaturowy: *Shift + Control + N*.

Uwaga. Domyślnie granice tekstu na stronie, widoczne na Rys. 12 w postaci linii przerywanej, nie są widoczne podczas edycji dokumentu. By stały się widoczne, trzeba albo wywołać makro znajdujące się na Quick Access Toolbar (patrz Rys. 4), albo zmienić ustawienie w opcjach aplikacji Microsoft Word.



Show text boundaries and text shading

Rys. 4. Włączanie / wyłączanie granicy tekstu na stronie.

* + 1. Krok 1: nadanie akapitom listy właściwego stylu „Połączone (akapit i znak)”

Przypiszmy pierwszemu i drugiemu akapitowi styl *ListaSeq 2 ms*.

Nadanie stylu dla zaznaczonych akapitów można osiągnąć na różne sposoby, na przykład:

* 1. wykorzystując skrót klawiatury *Alt + 2*, przypisany do stylu *ListaSeq 2 ms*,
  2. wykorzystując skrót klawiaturowy *Shift +Control + S* („s” jak „style”), wybór opcji „Autouzupełnianie nazw stylów” i w polu tekstowym wpisanie „ListaSe”, a następnie wybranie stylu o nazwie *ListaSeq 2 ms*.

Po przypisaniu tylko tego stylu Microsoft Word domyślnie, automatycznie, przypisze też do zaznaczonych akapitów styl typu „lista”. By tego uniknąć wystarczy usunąć numerację (przez naciśnięcie klawisza *Backspace (←)*). Po tej operacji powyższe dwa akapity wyglądają jak na Rys. 13, gdzie dla obu akapitów zostały ustawione wcięcia..



granica tekstu na stronie

Rys. 13. Wcięcie tekstu po nadaniu stylu *ListaSeq 2 ms*.

Widoczne jest m.in. „większe” wcięcie po lewej stronie, od granicy tekstu na stronie do granicy tekstu akapitów.

* + 1. Krok 2: nadanie akapitom listy właściwego stylu „Lista”

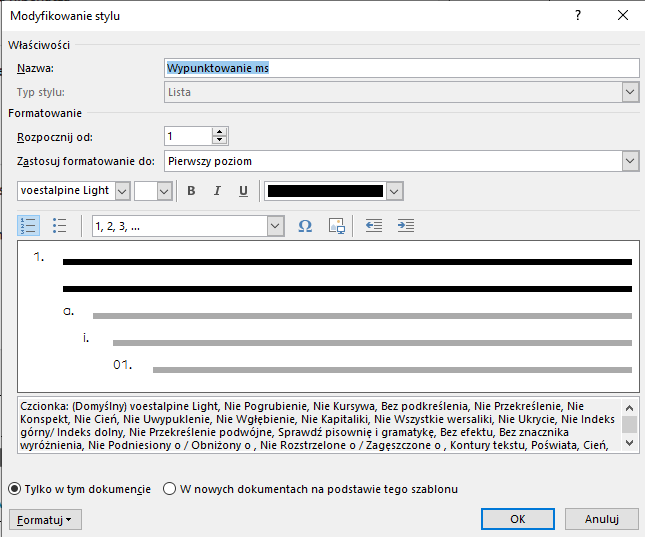
Pierwszemu i drugiemu akapitowi chcemy teraz przypisać styl *Wypunktowanie ms*. Nadanie stylu dla zaznaczonych akapitów można osiągnąć na różne sposoby, na przykład:

* 1. wykorzystując skrót klawiatury *Alt + L, W*, przypisany do stylu *Wypunktowanie ms*,
  + wykorzystując skrót klawiaturowy *Shift +Control + S* („s” jak „style”), wybór opcji „Autouzupełnianie nazw stylów” i w polu tekstowym wpisanie „Wyp”, a następnie wybranie stylu o nazwie *Wypunktowanie ms*.



Rys. 14. Po nadaniu stylu Wypunktowanie *ms*.

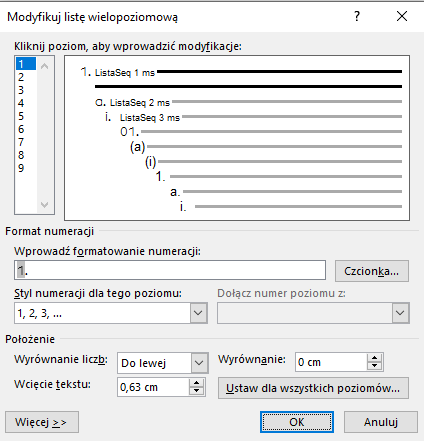
* 1. Jak powinna wyglądać konfiguracja stylu „Wypunktowanie ms”?



Rys. 15. Przykład.

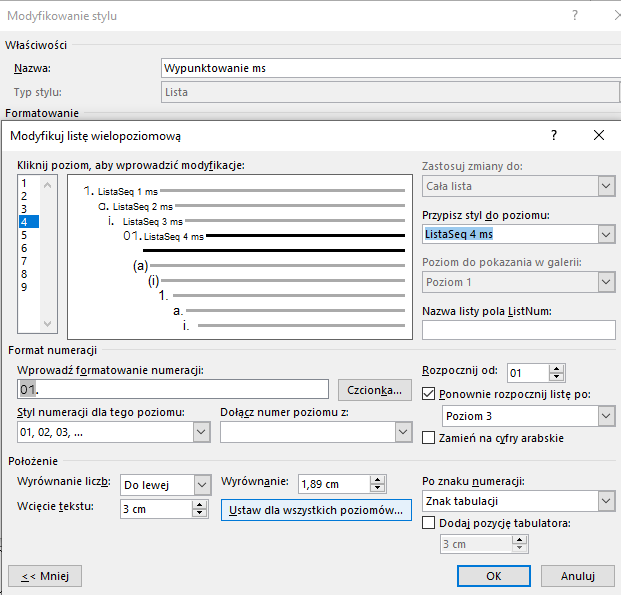
To musi być styl, którego typ jest określony jako „Lista”.

Wybrać przycisk Formatuj → Numerowanie…



Rys. 16. Przykład.

Wcisnąć przycisk Więcej >>

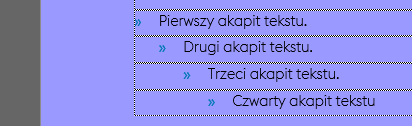
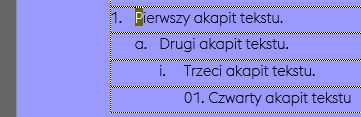


Rys. 17. Przykład.

Zadbać, by do poziomów 1 ÷ 4 zostały przypisane style *ListaSeq 1 ms* ÷ *ListaSeq 4 ms*.

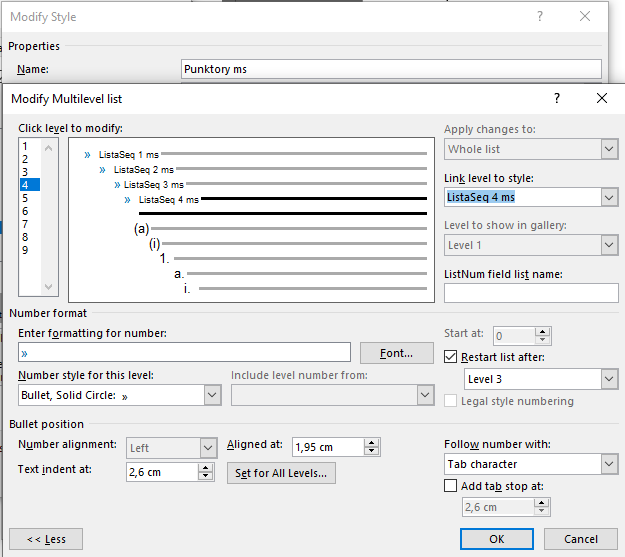
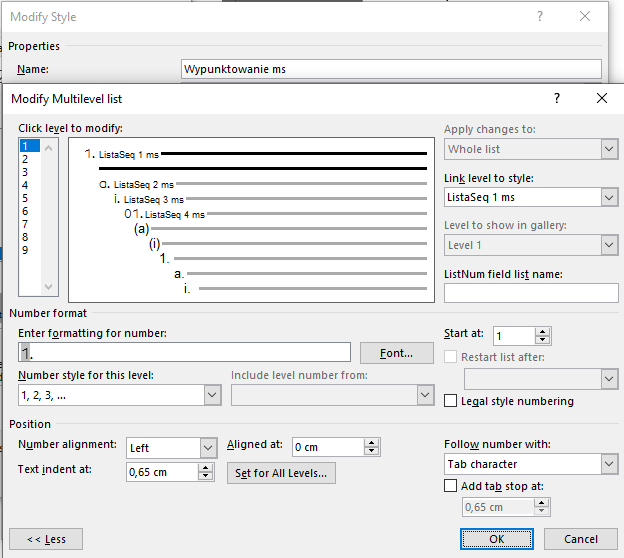
* + 1. Pułapki

Dobrym pomysłem wydaje się minimalizacja liczby styli w szablonie (mniejsza liczba ustawień konfiguracyjnych ułatwia ich utrzymywanie). Skoro mamy dwa style typu lista: *Wypunktowanie ms* oraz *Punktory ms*, które mają takie same wcięcia, a różnią się tylko sposobem oznaczania wypunktowań (porównaj Rys. 18), to wydaje się, że dobrym pomysłem będzie przypisanie tych samych styli typu „akapit i znak” do tych samych poziomów obu styli typu lista. Przykładowo, dla pierwszego poziomu stylu typu „lista” o nazwie *Wypunktowanie ms* i dla pierwszego poziomu stylu typu „lista” o nazwie *Punktory ms* przypisanie tego samego stylu typu akapit i znak *ListaSeq 1 ms.*



Rys. 18.Porównanie listy opartej na stylu *Wypunktowanie ms* (górna) z listą *Punktory ms*.

W praktyce jest to niemożliwe. To znaczy, że styl typu „akapit i znak” może zostać przypisany tylko do jednego (unikalnego) stylu typu „lista”. Użytkownik nie jest o tym informowany. Przy próbie przypisania tego samego stylu typu „akapit i znak” do innego stylu typu „lista” nie pojawi się żaden komunikat, a zadziała ostatnie przypisanie. Jest jedno z dziwactw aplikacji Microsoft Word.



Rys. 19. Nie jest możliwe przypisanie tych samych styli do poziomów różnych list.

Co zaskakujące mimo to poziomy wcięć oraz sposób nadawania wypunktowania zadziała poprawnie przy nadawaniu stylów typu „lista” akapitom tekstu dla obu styli typu „lista” (tak dla *Wypunktowanie ms* jak i dla *Punktory ms*). Dlaczego tak się dzieje, nie wiadomo.

Jeżeli chcielibyśmy poprawnie i jawnie skonfigurować oba omawiane style typu „lista” dla 4 poziomów listy, powinniśmy utworzyć drugi zestaw 4x styli typu „akapit i znak”.

* 1. Jak powiązać pola z wypunktowaniami?

Jak już wiemy z treści, od której rozpoczyna się rozdział 16, na listę składa się wcięcie akapitów treści oraz właściwe wypunktowanie (np. a, b, c).

Zamiast list wypunktowanych automatycznie przez Microsoft Word numeracje pól można nadać ręcznie, w oparciu o tzw. pola SEQ i bloki konstrukcyjne:

Definiujemy pole.

Definiujemy blok konstrukcyjny.

* + 1. Definiujemy pole SEQ

Używamy skrótu *Control + F9*. Następnie pomiędzy znakami „{” oraz „}” wpisujemy tekst jak na Rys. 13.



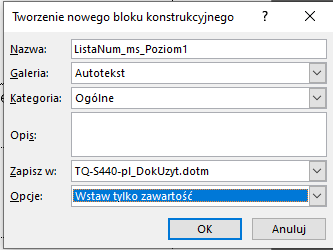
Rys. 20. Przykład.

Następnie wciskamy skrót *Alt + F9* (przełącz kody pól), zaznaczamy miejsce / akapit, gdzie znajduje się pole i wciskamy *F9*.

Należy pamiętać, że inaczej definiuje się pola, które rozpoczynają listy numerowane, a inaczej pola, które mają kontynuować numerowanie. Przykładowo, przełącznik „\r1” służy do rozpoczynania numeracji (resetowania), rozpoczynając od wartości 1.

* + 1. Definiujemy blok konstrukcyjny

Po wcześniej zdefiniowanym polu wstawiamy znaki „.” (kropka) oraz „⭾” (tabulator). Zaznaczamy sekwencję znaków bez zaznaczenia znaku końca akapitu „¶”. Następnie wybieramy z menu Wstawianie → Szybkie części → Zapisz zaznaczenie w galerii szybkich styli.



Rys. 21. Przykład.

Kluczowa sprawa to wybór „Galeria:” Autotekst oraz zapisanie bloku konstrukcyjnego w szablonie. Warto sprawdzić, czy tak utworzony blok konstrukcyjny rzeczywiście działa.

Przez to, że blok konstrukcyjny należy do kategorii „Autotekst”, powinien dać się wywołać za pomocą nazwy za pomocą aplikacji Hotstrings, a nawet bezpośrednio przez wprowadzanie tekstu w dokumencie, który odpowiada nazwie tego bloku konstrukcyjnego. Do zamiany tekstu na blok konstrukcyjny służy klawisz „↵” (Enter), ewentualnie F3, patrz Rys. 15.



Rys. 22. Przykład, wprowadzanie bloku konstrukcyjnego należącego do kategorii Autotekst.

Można pójść jeszcze jeden krok dalej i zdefiniować kolejny blok konstrukcyjny, który będzie zawierał cały akapit tekstu (ze znakiem końca akapitu ¶) i zawierał zarówno zdefiniowane pole wypunktowania jak i styl typu „Połączone (akapit i znak)”.

|  |
| --- |
| Wskazówka |
| Do szablonu należy wstawić zarówno bloki konstrukcyjne  A. zawierające tylko numerowanie  B. bloki konstrukcyjne zawierające kombinacje numerowania i stylu typu „Połączone (akapit i znak)”  C. Blok konstrukcyjny zawierający pełną listę do 4 poziomów zagnieżdżenia włącznie. |

* 1. Zasady stosowane dla list numerowanych i nienumerowanych w szablonach

Maksymalnie 4 poziomy.

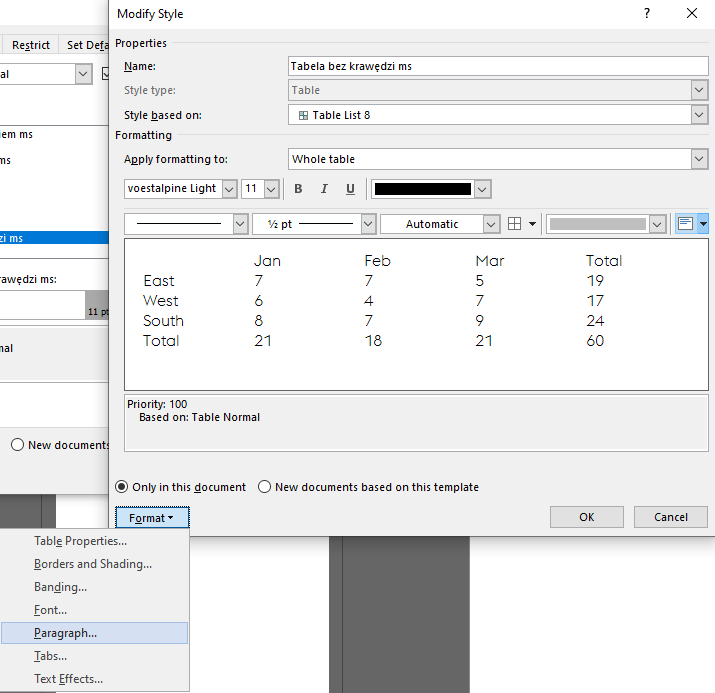
Numerowane mają priorytet nad nienumerowanymi.   
Numerowane umożliwiają precyzyjne odwołanie się do treści.

Tab. 1. Zasady dopasowania wyrównania i wcięć.

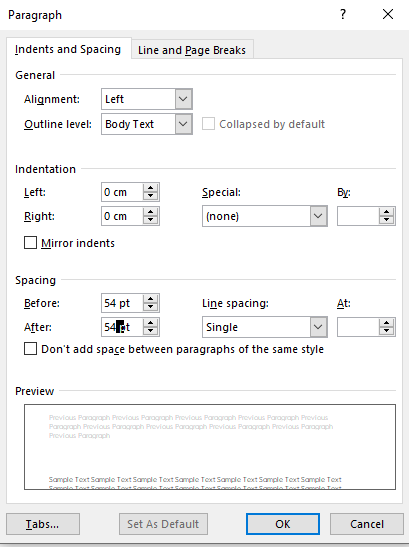
|  |  |  |
| --- | --- | --- |
|  | wcięcia z lewej [cm] | specjalne [cm] |
| poziom 1 | 0 | wysunięcie 0,65 (= 0 + 0,65) |
| poziom 2 | 0,65 (wartość wcięcia poprzedniego poziomu) | wysunięcie 1,3 (= 0,65 + 0,65) |
| poziom 3 | 1,3 (wartość wcięcia poprzedniego poziomu) | wysunięcie 1,95 (= 1,3 + 0,65) |
| poziom 4 | 1,95 (wartość wcięcia poprzedniego poziomu) | wysunięcie 2,6 (= 1,95 + 0,65) |

1. Tabele

„Tabele” w Microsoft Word to także typ stylu. Wyróżnia ten typ między innymi to, że nie można przypisać do niego odstępów akapitów przed tabelą i po tabeli, pomimo że teoretycznie jest to możliwe, bo takie ustawienia są aktywne w oknie „zarządzanie stylami” (patrz Rys. 23). Pomimo nadania nawet znacznych wartości odstępom dla paragrafu poprzedzającego i następującego, nie zobaczymy różnicy, bo ustawienia dotyczą paragrafu tekstu umieszczonego w komórkach tabeli, a nie paragrafu, w którym umieszczona została tabela.

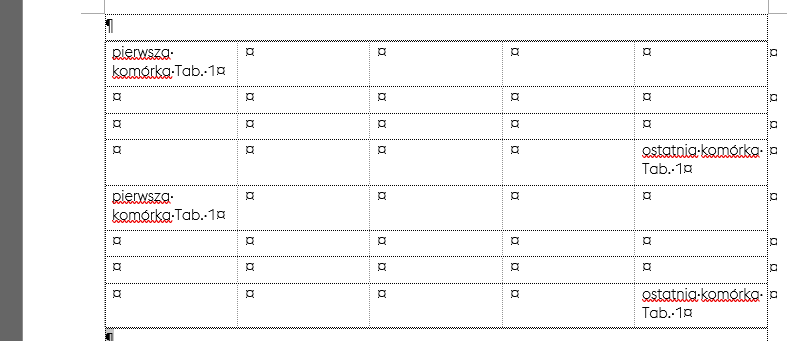


Rys. 23. Próba modyfikacji właściwości paragrafu dla stylu „tabela”.



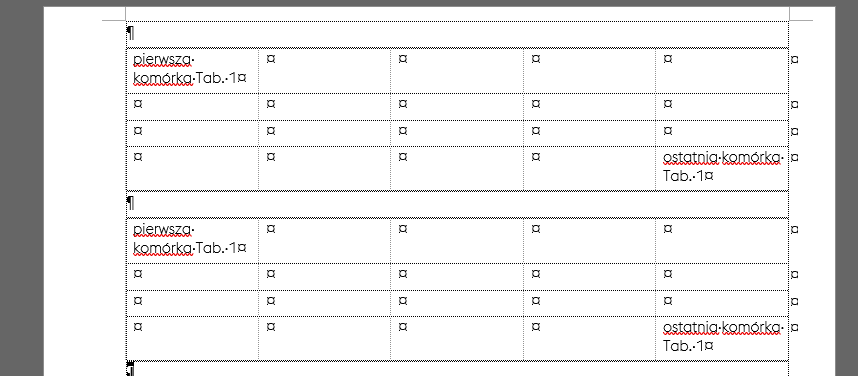
Rys. 24. Zmiana ustawień wartości odstępów dla poprzedniego i kolejnego akapitu.

Konsekwencją tej cechy jest „zlewanie się” tabel umieszczonych jedna pod drugą.



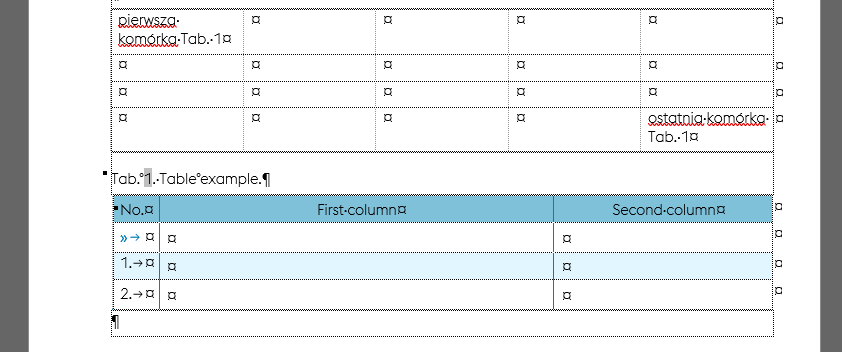
Rys. 25. Dwie tabele połączyły się w jedną.

Jedynym rozwiązaniem w takiej sytuacji jest wstawienie pustego akapitu ¶, co z kolei może powodować problemy z łamaniem się strumienia tekstu przy podziale na strony.



Rys. 26. Pusty paragraf wykorzystany do rozdzielenia tabel.

Wniosek: należy unikać połączeń składających się z kolejno następujących po sobie tabel. Stosunkowo łatwo jest uniknąć takiej sytuacji, o ile na przykład przyjmie się zasadę, że każda tabela powinna być poprzedzona etykietą, czyli dodatkowym akapitem tekstu.



Rys. 27. Dwie tabele rozdzielone akapitem z etykietą tabeli.

* 1. Jak formatować tabele?

Formatowanie tabel polega na nadaniu przynajmniej dwóch styli:

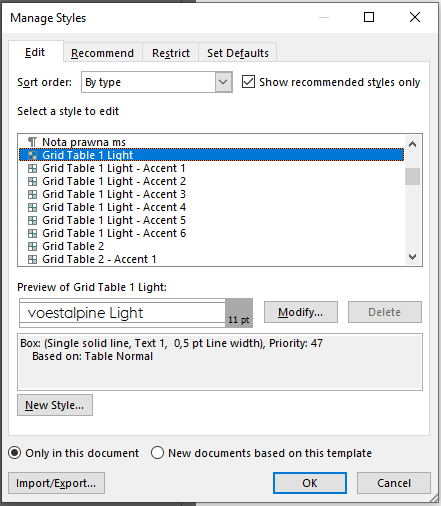
* 1. stylu typu „tabela”, który pokoloruje kolumny, ustawi ich wyrównanie, ustawi widoczność krawędzi,
  2. stylu typu „akapit i znak”, który pozwoli na ustawienie cech akapitów w każdej z komórek tabeli,
  3. opcjonalnie trzeci styl, który pozwoli na zapanowanie nad numeracją styli w wybranej kolumnie / kolumnach.

Formatowanie pojedynczej tabeli wymaga więc przynajmniej dwukrotnego zastosowania styli.

Niestety stylom typu „tabela” nie można przypisać skrótów klawiaturowych, co powoduje, że formatowanie tabel wymaga dodatkowego nakładu pracy.

* 1. Style typu „tabela” a szablon

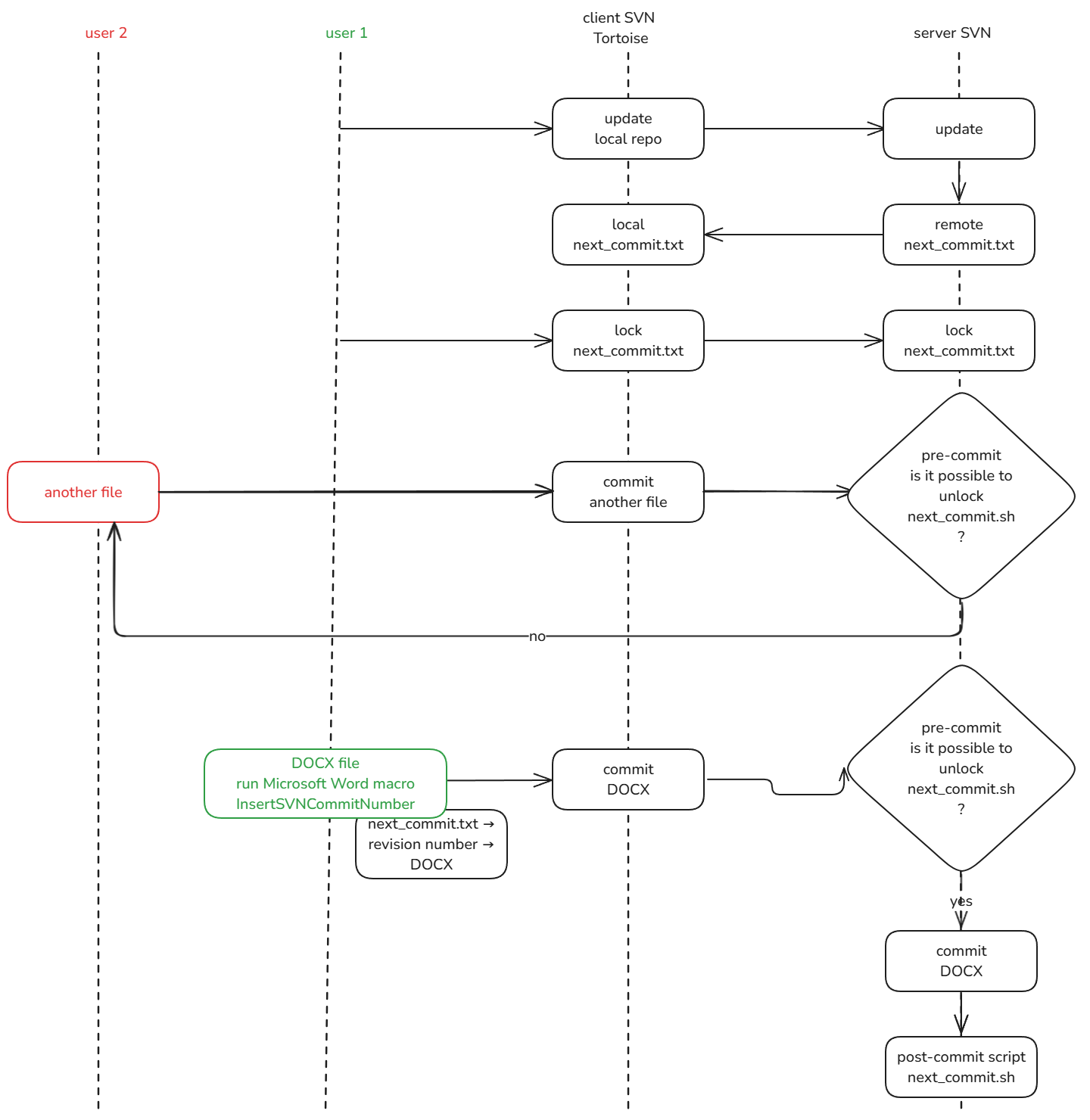
Z plików szablonu niestety nie da się usunąć wbudowanych styli typu „tabela”, a jest ich sporo.



Rys. 28. Przykład stylu wbudowanego, którego nie można usunąć z szablonu.

1. SVN

Cel: Umieszczenie nr rewizji SVN w polu danych identyfikacyjnych dokumentu.



Pic. 52. Rozwiązanie, rewizja SVN. [source: 20250401\_SVN\_DOCX.excalidraw]

Wykorzystać mechanizm haków (ang. hooks) po stronie serwera, do tworzenia pliku next\_commit.txt, tworzonego i utrzymywanego w głównym folderze repozytorium jako operacja post-commit. Plik ten powinien zawierać następny nr rewizji (następny względem aktualnego). Plik jest aktualizowany automatycznie, przy każdej operacji aktualizacji treści w repozytorium (ang. commit). Skrypt hak pre-commit sprawdza dla każdej innej operacji aktualizacji treści serwera (ang. commit), czy zablokowany (ang. lock) jest plik next\_commit.txt. Jeżeli ten plik jest zablokowany, to uniemożliwia aktualizację.

* 1. Kod skryptu bash po stronie serwera post-commit.sh

| **⚠** Warning |
| --- |
| Nie przetestowany. |

#!/bin/bash

REPOS="$1"

TXN="$2"

# Path to the file where the next commit number will be stored

NEXT\_COMMIT\_FILE="/path/to/next\_commit.txt"

# Generate a unique identifier for the next commit

NEXT\_COMMIT=$(svnlook youngest $REPOS)

NEXT\_COMMIT=$((NEXT\_COMMIT + 1))

# Write the next commit number to the file

echo "$NEXT\_COMMIT" > $NEXT\_COMMIT\_FILE

# Allow the commit to proceed

exit 0

Ten skrypt powinien zostać umieszczony na serwerze w folderze hooks.

Podczas aktualizacji lokalnej kopii repozytorium (ang. update), w lokalnym repozytorium użytkownika znajdzie się kopia pliku next\_commit.txt z nr kolejnej rewizji.

* 1. Kod skryptu bash po stronie serwera pre-commit.sh

| **⚠** Warning |
| --- |
| Nie przetestowany. |

Ten skrypt powinien zostać umieszczony na serwerze w folderze hooks.

#!/bin/bash

REPO\_PATH="/path/to/repository"

LOCK\_FILE="$REPO\_PATH/next\_commit.lock"

COMMIT\_FILE="$REPO\_PATH/next\_commit.txt"

# Check if the lock file exists

if [ -f "$LOCK\_FILE" ]; then

echo "Error: next\_commit.txt is locked. Please try again later."

exit 1

fi

# Create the lock file

touch "$LOCK\_FILE"

# Perform the commit

svn commit -m "$1"

# Remove the lock file

rm "$LOCK\_FILE"

exit 0

* 1. Kod makra VBA InsertSVNCommitNumber

Nr rewizji z pliku next\_commit.txt należy umieścić w aktualnie otwartym pliku DOCX. Można to zrobić za pomocą makra VBA InsertSVNCommitNumber (szablon: ???).

Sub InsertSVNCommitNumber()

Dim CommitFilePath As String

Dim CommitNumber As String

Dim CommitFile As Integer

Dim doc As Document

Dim rng As Range

Dim UserResponse As VbMsgBoxResult

Const CommitFilename As String = "next\_commit.txt"

Const DocumentVariableName As String = "DV\_CommitFilePath"

Const DocCP\_SVN\_Revision As String = "ms\_SVN\_Revision" ' Document Custom Property

Dim macroName As String

Dim moduleName As String

Dim MsgBoxHeader As String

macroName = "InsertSVNCommitNumber"

If Not Application.VBE.ActiveCodePane Is Nothing Then

moduleName = Application.VBE.ActiveCodePane.CodeModule.Name

End If

MsgBoxHeader = macroName & " in module " & moduleName

' Initialize commitFilePath with Document Variable DV\_CommitFilePath

On Error Resume Next

CommitFilePath = ActiveDocument.Variables(DocumentVariableName).value

On Error GoTo 0

' Check if DV\_CommitFilePath is empty

If CommitFilePath = "" Then

' Display dialog for user to select folder containing next\_commit.txt

With Application.FileDialog(msoFileDialogFolderPicker)

.Title = "Select folder containing " & CommitFilename

If .Show = -1 Then

CommitFilePath = .SelectedItems(1) & "\" & CommitFilename

ActiveDocument.Variables(DocumentVariableName).value = CommitFilePath

Else

MsgBox "No folder selected. Operation cancelled.", \_

vbInformation, MsgBoxHeader

Exit Sub

End If

End With

Else

' Display MsgBox with specified DV\_CommitFilePath and ask if user wishes to change it

UserResponse = MsgBox("Current commit file path: " & CommitFilePath & vbCrLf & "Do you wish to change it?", \_

vbYesNo + vbQuestion, MsgBoxHeader)

If UserResponse = vbYes Then

' Display dialog for user to select folder containing next\_commit.txt

With Application.FileDialog(msoFileDialogFolderPicker)

.Title = "Select folder containing " & CommitFilename

If .Show = -1 Then

CommitFilePath = .SelectedItems(1) & "\" & CommitFilename

ActiveDocument.Variables(DocumentVariableName).value = CommitFilePath

Else

MsgBox "No folder selected. Operation cancelled.", \_

vbInformation, MsgBoxHeader

Exit Sub

End If

End With

End If

End If

' Open the text file and read the commit number

CommitFile = FreeFile

Open CommitFilePath For Input As CommitFile

Input #CommitFile, CommitNumber

Close CommitFile

' Store the commit number in a document custom property named ms\_SVN\_Revision

Set doc = ActiveDocument

doc.CustomDocumentProperties(DocCP\_SVN\_Revision).value = CommitNumber

Call UpdateAllFields ' module: Validation

Set doc = Nothing

MsgBox prompt:="Document custom property '" & DocCP\_SVN\_Revision & "' was updated with SVN revision no. " & CommitNumber & "." & vbNewLine & \_

"All fields in document have been updated.", \_

buttons:=vbInformation, Title:=MsgBoxHeader

End Sub

W celu wykluczenia konfliktów należy na czas wykonywania operacji na pliku DOCX zamknąć dostęp do pliku next\_commit.txt za pomocą mechanizmu SVN lock. W ten sposób, dopóki nie zostanie wykonany kolejny zapis do repozytorium na serwerze (ang. commit), inni użytkownicy nie będą w stanie zmodyfikować tego pliku.

1. This book repository

https://svn.tens.pl/CompanyTemplates/UnofficialTemplates

* 1. Structure

Struktura folderów i plików przystaje do Pic. 1.

1. Backup of working files
2. Jak szukać pomocy do Microsoft Word?

Poniżej podaję oficjalne metody.

* 1. Oficjalna pomoc online

<https://support.microsoft.com/pl-PL/microsoft-365>

* 1. Oficjalne forum

<https://answers.microsoft.com/en-us/msoffice/forum>

Przykłady wyzwań:

[How to run automatically Document\_New macro with /t command line - Microsoft Community](https://answers.microsoft.com/en-us/msoffice/forum/all/how-to-run-automatically-documentnew-macro-with-t/e3ba6e61-a8ff-4533-8c7d-b6de5a205995?messageId=14849a95-6bd3-47c6-a675-219aacb1fd11)

* 1. Zgłaszanie nieprawidłowości w Microsoft Word

<https://feedbackportal.microsoft.com/feedback>

* 1. Official releases and known bugs

[Fixes or workarounds for recent issues in Word for Windows - Microsoft Support](https://support.microsoft.com/en-us/office/fixes-or-workarounds-for-recent-issues-in-word-for-windows-bf6bf17c-2807-4871-83ce-e337ae8f0b86)

[Update history for Microsoft 365 Apps (listed by date) - Office release notes | Microsoft Learn](https://learn.microsoft.com/en-us/officeupdates/update-history-microsoft365-apps-by-date)