Document Object Model

Jarosław Kuchta

Ten dokument opisuje obiektowy model dokumentu (DOM – Document Object Model) skonstruowany na podstawie istniejących modeli dokumentu programu Word. Celem projektu było opracowanie modelu, który umożliwia przetwarzanie dokumentów Worda zapisanych w pliku DOCX bez dostępu do aplikacji Worda. Model ten jest wymagany przez program konwersji z DOCX do formatu LaTeX, lecz może też być wykorzystywany w innych aplikacjach przetwarzających dokumenty Word, również w aplikacjach uruchamianych nie tylko pod Windows, ale też w innych systemach operacyjnych. Wystarczy do tego zgodność aplikacji z .NET Standard 2.0.

# Typy podstawowe

Typy podstawowe ułatwiają dopasowanie projektu modelu obiektowego dokumentu do jego pierwowzorów.

## Typ wariantowy

Typ Variant wykorzystuje się we właściwościach CustomDocumentProperties, a także niektórych właściwościach ContentDocumentProperties.

W przestrzeni nazw DocumentFormat.OpenXml.VariantTypes, zdefiniowane są typy wariantowe przedstawione tu w tab. 1. Dwa z pośród nich to typu złożone: VTArray i VTVector, które mogą zawierać wiele elementów typu wariantowego. Element typu Variant może zawierać inny element wariantowy dowolnego typu.

Tab. . Typy wariantowe w OpenXML

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Typ | Zawartość | El. OpenXML | El. tablicy | El. wektora |
| Variant | Variant | vt:variant | • | • |
| VTArray | Array | vt:array |  |  |
| VTBlob | Binary Blob | vt:blob |  |  |
| VTBool | bool | vt:bool | • | • |
| VTBString | Basic String | vt:bstr |  | • |
| VTByte | 1-Byte Signed Integer | vt:i1 | • | • |
| VTClassId | Class ID | vt:clsid |  | • |
| VTClipboardData | Clipboard Data | vt:cf |  |  |
| VTCurrency | Currency | vt:cy | • | • |
| VTDate | Date and Time | vt:date | • | • |
| VTDecimal | Decimal | vt:decimal | • |  |
| VTDouble | 8-Byte Real Number | vt:r8 | • | • |
| VTEmpty | Empty | vt:empty |  |  |
| VTError | Error Status Code | vt:error | • | • |
| VTFileTime | File Time | vt:filetime |  | • |
| VTFloat | 4-Byte Real Number | vt:r4 | • | • |
| VTInt32 | 4-Byte Signed Integer | vt:i4 | • | • |
| VTInt64 | 8-Byte Signed Integer | vt:i8 |  | • |
| VTInteger | Integer | vt:int | • |  |
| VTLPSTR | LPSTR | vt:lpstr |  | • |
| VTLPWSTR | LPWSTR | vt:lpwstr |  | • |
| VTNull | Null | vt:null |  |  |
| VTOBlob | Binary Blob Object | vt:oblob |  |  |
| VTOStorage | Binary Storage Object | vt:ostorage |  |  |
| VTOStreamData | Binary Stream Object | vt:ostream |  |  |
| VTShort | 2-Byte Signed Integer | vt:i2 | • | • |
| VTStorage | Binary Storage | vt:storage |  |  |
| VTStreamData | Binary Stream | vt:stream |  |  |
| VTUnsignedByte | 1-Byte Unsigned Integer | vt:ui1 | • | • |
| VTUnsignedInt32 | 4-Byte Unsigned Integer | vt:ui4 | • | • |
| VTUnsignedInt64 | 8-Byte Unsigned Integer | vt:ui8 |  | • |
| VTUnsignedInteger | Unsigned Integer | vt:uint | • |  |
| VTUnsignedShort | 2-Byte Unsigned Integer | vt:ui2 | • | • |
| VTVector | Vector | vt:vector |  |  |
| VTVStreamData | Binary Versioned Stream | vt:vstream |  |  |

Niektóre typy wymagają komentarza:

* Typ Variant może wystąpić tylko w tabeli lub w wektorze i może zawierać dowolny z typów wariantowych. Przydaje się, gdy trzeba umieścić w tablicy lub wektorze element, którego typ nie jest dozwolony w kontekście tablicy lub wektora.
* Typ VTArray zawiera tablicę elementów jednolitego typu. Dozwolone typy elementów są wymienione w kolumnie „El. tablicy”. Tablica może być wielowymiarowa i mieć określone dolne i górne granice indeksów w każdym wymiarze.

Przykład

<vt:array lBounds="0,0" uBounds="1,2" baseType="i4">

<vt:i4>0</vt:i4>

<vt:i4>1</vt:i4>

<vt:i4>2</vt:i4>

<vt:i4>3</vt:i4>

<vt:i4>4</vt:i4>

</vt:array>

Wynikiem jest tabela [0,0] = 0, [1,0] = 1, [0,1] = 2, [1,1] = 3, [0,2] = 4.

* Typ VTVector jest jednowymiarową tablicą elementów określonego typu. Dozwolone typy elementów są wymienione w kolumnie „El. wektora”.

Przykład

<vt:vector baseType="lpstr" size="3">

<vt:lpstr>One</vt:lpstr>

<vt:lpstr>Two</vt:lpstr>

<vt:lpstr>Three</vt:lpstr>

</vt:vector>

* Typy VTBString, VTLPSTR i VTLPWSTR są implementowane tak samo, jako łańcuch znaków Unicode. Znaki spoza zakresu ASCII mogą być w XML zakodowane jako \_xHHHH\_, gdzie H oznacza cyfrę szesnastkową.
* Typy VTBlob, VTOBlob, VTOStorage, VTOStreamData, VTStorage, VTStreamData i VTVStreamData kodują dane binarne w postaci ciągu base64Binary. VTVStreamData ma określony typ zawartości przez GUID.
* Typ VTClassId identyfikuje klasę przez GUID.
* Typ VTClipboardData ma zawartość określoną przez atrybut liczbowy format.
* Typ VTError podaje 32-bitowy kod błędu w postaci 0xHHHHHHHH.

Na podstawie tego zdefiniowano typ strukturalny Variant oraz typ wyliczeniowy VariantType. W VariantType zdefiniowano nieco mniejszy zbiór rozróżnialnych typów wariantowych (kilka typów wariantowych z tab. 1 odwzorowano w te same wartości wyliczeniowe).

Tab. . Typy wariantów

|  |  |  |
| --- | --- | --- |
| Val | VariantType | Value |
| 0 | Empty | null |
| 1 | Null | null |
| 2 | Bool | bool |
| 3 | Int8 | SByte |
| 4 | Int16 | Int16 |
| 5 | Int32 | Int32 |
| 6 | Int64 | Int64 |
| 7 | UInt8 | Byte |
| 8 | UInt16 | UInt16 |
| 9 | UInt32 | UInt32 |
| 10 | UInt64 | UInt64 |
| 11 | Float | Single |
| 12 | Double | Double |
| 13 | Decimal | Decimal |
| 14 | Currency | Decimal |
| 15 | DateTime | DateTime |
| 16 | String | String |
| 17 | Blob | byte[] |
| 18 | OBlob | byte[] |
| 19 | Storage | byte[] |
| 20 | OStorage | byte[] |
| 21 | StreamData | byte[] |
| 22 | OStreamData | GuidDataPair |
| 23 | VStreamData | GuidDataPair |
| 24 | ClassId | Guid |
| 25 | Error | Int32 |
| 26 | ClipboardData | ClipboardData |
| 0x0FFF | Variant | Variant |
| 0x2000 | Array | Array |
| 0x4000 | Vector | ICollection |

Wartości wyliczeniowe dla typów Variant, Array i Vector ustalono tak, aby można było za pomocą maskowania zapisać typy elementów składowych. Reprezentacje wartości dla typów Blob, OBlob, Storage, OStorage, StreamData OStreamData, VStreamData są niepewne ze względu na słabą dokumentację i brak dostępnych przykładów. Z tego samego względu brak pełnej implementacji dla typu ClipboardData. Ze względu na ograniczenie implementacji OpenXML typ Array obsługuje tylko tablice jednowymiarowe.

Typ Variant ma cztery właściwości (tab. 3), z czego dwie służą do przechowywania danych, a dwie służą tylko do odczytu.

Tab. . Właściwości wariantu

|  |  |  |  |
| --- | --- | --- | --- |
| Właściwość | Typ | R/W | Znaczenie |
| Type | VariantType | R/W | Typ wariantu |
| Value | Object? | R/W | Wartość |
| IsArray | bool | R/O | Czy wariant jest tablicą? |
| IsCollection | bool | R/O | Czy wariant jest wektorem? |

## Kolekcje elementów

Zdefiniowano trzy typy kolekcji:

* Collection<ItemType> – uogólniony typ kolekcji dowolnych elementów,
* Collection<KeyType, ItemType> –typ indeksowanej kolekcji elementów,
* Strings – szczególny typ kolekcji łańcuchów znakowych.

Typ Collection<ItemType> reprezentuje kolekcję elementów podanego typu. Oferuje możliwości wyszczególnienia kolejnych elementów, dodania i usunięcia elementu. Indeksowanie elementów jest po liczbach całkowitych, gdzie pierwszy element ma indeks 0.

Przykładem takiej kolekcji jest typ Strings, który jest prostą kolekcją łańcuchów znakowych: Collection<string>.

Typ Collection<KeyType, ItemType> reprezentuje kolekcję elementów indeksowanych podanego typu, gdzie KeyType reprezentuje typ klucza. Typ elementu powinien mieć jedną z właściwości oznaczoną atrybutem [Key]. Ten typ kolekcji daje możliwość enumerowania po elementach w kolejności ich dodawania.

## Typy liczbowe

Typy liczbowe definiują jednostki, w których są wyrażone ich wartości. Umożliwiają konwersję na inną jednostkę.

Tab. . Typy liczbowe

|  |  |  |
| --- | --- | --- |
| Nazwa typu | Znaczenie | Uwagi |
| Percentage | Typ stałoprzecinkowy reprezentujący wartość procentową | Konwersja na typ double daje wartość w skali 100%=1.0 |
| Twips | Typ całkowity reprezentujący wartość w 1/1440 cala | Konwersja na typ double daje wartość w calach |

## Tablice bajtów

Zdefiniowano dwa typy tablic bajtowych, które mają wartość typu byte[], a różnią się sposobem konwersji na łańcuch znaków.

Tab. . Tablice bajtów

|  |  |  |
| --- | --- | --- |
| Nazwa typu | Znaczenie | Uwagi |
| HexBinary | Tablica bajtów wyrażana przez łańcuch dwucyfrowych liczb szesnastkowych | Przeznaczony do krótkich łańcuchów. Ma konwersję na typy byte, ushort, int, uint. |
| Base64Binary | Tablica bajtów wyrażana przez łańcuch znaków kodowany w standardzie base64 | Przeznaczony do dłuższych łańcuchów. |

## Typy wspólne

Typy zadeklarowane w tej grupie są używane w różnych kontekstach dokumentu. Zestawienie jest przedstawione w tab. 6.

Tab. . Typy wspólne

|  |  |  |
| --- | --- | --- |
| Nazwa typu | Znaczenie | Uwagi |
| BorderDistanceFrom | Określa, czy obramowanie strony jest mierzone od krawędzi strony, czy od otaczającego ją tekstu. | Wykorzystywany w klasie PageBorders. |
| ClipboardData | Definiuje dane przekazywane przez schowek (wraz z formatem) | Wykorzystywany w typie Variant. |
| Color | Definiuje kolor jako strukturę 3 bajtów RGB. Wartość może być zastąpiona przez nazwę. | Wykorzystywany wielokrotnie. |
| ColorFormat | Definiuje kolor jako RGB, CMYK lub systemową nazwę. | Wykorzystywany wielokrotnie. |
| ColorSchemeMapping | Definiuje odwzorowanie kolorów schematu. | Wykorzystywany w ustawieniach |
| ColorIndex | Definiuje kolor jako wartość indeksowaną od 0 do 16, w tym: -1 oznacza kolor definiowany przez autora, 0 – automatyczny, 1 – czarny, 8- biały, 15- szary 50%, 16 – szary 25%. Pozostałe to kolory podstawowe RGB i mieszane. | Wykorzystywany wielokrotnie. |
| ColorType | Definiuje rodzaj koloru: RGB, Scheme, CMYK, CMS, Ink lub Mixed. | Wykorzystywany w ColorFormat. |
| EmphasisMark | Definiuje rodzaj znaku nacisku: kropka umieszczana ponad znakiem lub pod znakiem, kółko lub przecinek ponad znakiem. | Wykorzystywany w RunProperties. |
| FrameBorders | Definiuje obramowanie w postaci ramki – wszystkie krawędzie mają ten sam styl i kolor. | Wspólna klasa bazowa dla RunBorders i PageBorders. |
| LanguageType | Definiuje języki dla pisma standardowego, dwukierunkowego i dalekowschodniego | Wykorzystywany w w ustawieniach |
| LineWidth | Definiuje kilka podstawowych szerokości linii | Wykorzystywany w FrameProperties. |
| NamedColors | Definiuje nazwy i kody kilkudziesięciu nazwanych kolorów. | Wykorzystywany w typie Color. |
| PageBorders | Reprezentuje obramowanie przypisane do strony |  |
| RunBorders | Reprezentuje obramowanie przypisane do ciągu tekstowego | Wykorzystywany w RunProperties. |
| Shading | Reprezentuje cieniowanie obiektu. Może mieć wzór, kolor tła i kolor wzoru. | Wykorzystywany w RunProperties. |
| ShadowFormat | Reprezentuje cień rzucany przez obiekt. Może mieć różny typ, styl, kolor, odległość, kąt i inne właściwości. | Wykorzystywany w RunProperties. |
| ShadowStyle | Definiuje styl cienia. Może być wewnętrzny, zewnętrzny lub mieszany. | Wykorzystywany w ShadowFormat. |
| ShadowType | Definiuje 43 predefiniowane typy cienia. | Wykorzystywany w ShadowFormat. |
| TextureIndex | Definiuje kilkanaście wzorów cieniowania. | Wykorzystywany w Shading. |
| ThemeColorIndex | Definiuje indeks koloru w motywie koloru. | Wykorzystywany w ColorFormat. |

# Dokument i jego struktura

## Właściwości dokumentu

W projekcie modelu dokumentu właściwości dokumentu są przechowywane w klasie DocumentProperties. Klasa ta implementuje 6 interfejsów:

* IDocumentProperties,
* ICoreDocumentProperties,
* IContentDocumentProperties,
* IStatisticDocumentProperties,
* IExtraDocumentProperties,
* ICustomDocumentProperties.

Interfejs IDocumentProperties reprezentuje kolekcję właściwości dokumentu, umożliwia ich dodawanie, zmianę i usuwanie oraz wyszukiwanie po nazwie i numerze indeksu (kolejności wprowadzania). Metoda Set umożliwia dodanie właściwości o podanej nazwie, a jeśli taka już istnieje, to jej zmianę. Metoda Get umożliwia pobranie właściwości o podanej nazwie, a jeśli taka nie istnieje, to zwraca wartość null. Wielkość liter w nazwach właściwości nie jest istotna.

Pojedyncza właściwość dokumentu jest reprezentowana przez klasę DocumentProperty, która implementuje interfejs IDocumentProperty. Ma ona trzy właściwości:

* Name: string – unikatowa nazwa właściwości (wielkość liter nie ma znaczenia),
* Type: PropertyType – typ właściwości (wg typu wyliczeniowego),
* Value: object? – wartość właściwości (zgodnie z typem), może być null.

Typy właściwości są określone przez typ wyliczeniowy PropertyType, który definiuje następujące typy wartości:

* Object – dowolny typ nie objęty przez pozostałe kategorie,
* Number – typ liczbowy całkowity (int, int64, decimal),
* Boolean – typ logiczny (bool),
* Date – typ daty i czasu (DateTime, DateOnly),
* String – typ znakowy (string),
* Float – typ rzeczywisty (float, double),

Interfejs ICoreDocumentProperties definiuje właściwości główne dokumentu, te które Microsoft Word przechowuje w części Core File Properties. Interfejs IContentDocumentProperties definiuje właściwości przechowywane przez MS Word w części Extended File Properties z wyjątkiem informacji o statystyce dokumentu, które są definiowane przez interfejs IStatisticDocumentProperties. Interfejs IExtraDocumentProperties definiuje dodatkowe właściwości dokumentu, które są przechowywane w części Document Settings.

Wszystkie te właściwości są przedstawione w tab. 7. Do każdej z nich można się odwołać przez indeksator klasy DocumentProperties podając jej nazwę.

Tab. . DocumentProperties

|  |  |  |
| --- | --- | --- |
| Właściwość | Typ | Znaczenie |
| **Core Document Properties** | | |
| Title | string? | Tytuł dokumentu |
| Subject | string? | Temat dokumentu. |
| Category | string? | Kategoria dokumentu. |
| ContentType | string? | Typ dokumentu. |
| ContentStatus | string? | Status dokumentu. |
| Description | string? | Krótki opis dokumentu. |
| Keywords | string? | Rozdzielany przecinkami lub średnikami zestaw słów kluczowych. |
| Creator | string? | Autor, który utworzył dokument. |
| Created | DateTime? | Data utworzenia dokumentu. |
| LastModifiedBy | string? | Użytkownik, który wykonał ostatnią modyfikację. |
| LastModified | DateTime? | Data ostatniej zmiany zasobu. |
| LastPrinted | DateTime? | Data i godzina ostatniego wydruku. |
| Language | string? | Język treści, np. kodowany w standardzie IETF 3066. |
| Identifier | string? | Jednoznaczne odniesienie do zasobu w danym kontekście. |
| Version | string? | Numer wersji zasobu. Ustawiana przez użytkownika lub przez aplikację. |
| Revision | string? | Numer wersji zasobu. |
| **Content Document Properties** | | |
| Application | string? | Nazwa aplikacji, która utworzyła ten dokument. |
| ApplicationVersion | string? | Wersja aplikacji, która wyprodukowała ten dokument. Treść tego elementu ma postać XX.RRRR, gdzie X i Y reprezentują wartości liczbowe, lub dokument uznaje się za niezgodny. |
| Company | string? | Nazwa firmy skojarzonej z dokumentem. |
| DigitalSignature | byte[]? | Podpis cyfrowy dokumentu (właściwość zostawiona dla zgodności ze starszymi wersjami aplikacji). |
| DocumentSecurity | DocSecurity? | Poziom zabezpieczeń dokumentu. |
| HeadingPairs | HeadingPairs? | Pary nagłówków wskazują grupowanie części dokumentu i liczbę części w każdej grupie. Te części są koncepcyjnymi reprezentacjami sekcji dokumentu. |
| HyperlinkList | HyperlinkList? | Zestaw hiperłączy, które znajdowały się w tym dokumencie podczas ostatniego zapisywania. |
| HyperlinkBase | string? | Ciąg podstawowy używany do rozwiązywania hiperłączy względnych w tym dokumencie. |
| HyperlinksChanged | bool? | Ten element określa, że jedno lub więcej hiperłączy w tej części zostało zaktualizowanych wyłącznie w tej części przez producenta. Następny producent, który otworzy niniejszy dokument, powinien zaktualizować relacje hiperłączy o nowe hiperłącza określone w niniejszej części. |
| LinksUpToDate | bool? | Ten element wskazuje, czy hiperłącza w dokumencie są aktualne. |
| Manager | string? | Użytkownik nadzorujący pracę z dokumentem. |
| PresentationFormat | string? | Zamierzony format dokumentu prezentacji. |
| ScaleCrop | bool? | Ten element wskazuje tryb wyświetlania miniatury dokumentu. TRUE umożliwia skalowanie miniatury dokumentu do ekranu. FALSE umożliwia przycinanie krawędzi dokumentu w celu wyświetlenia tylko sekcji, które pasują do wyświetlacza. |
| SharedDocument | bool? | Ten element wskazuje, czy ten dokument jest obecnie współużytkowany przez wielu producentów. Jeśli ten element jest ustawiony na TRUE, producenci powinni zachować ostrożność podczas aktualizacji dokumentu. |
| Template | string? | Nazwa zewnętrznego szablonu dokumentu zawierającego informacje o formacie i stylu użyte do utworzenia bieżącego dokumentu. |
| TitlesOfParts | TitlesOfPairs? | Tytuł każdej części dokumentu. Te części są koncepcyjnymi reprezentacjami sekcji dokumentu. |
| **Statistic Document Properties** | | |
| TotalTime | int? | Całkowity czas edycji dokumentu. Domyślną jednostką czasu są minuty. |
| Characters | int? | Całkowita liczba znaków w dokumencie. |
| CharactersWithSpaces | int? | Całkowita liczba znaków (ze spacjami) w tym dokumencie. |
| Words | int? | Całkowita liczbę wyrazów zawartych w dokumencie podczas ostatniego zapisywania. |
| Lines | int? | Całkowita liczba wierszy w dokumencie, gdy ostatnio został zapisany przez zgodnego producenta, jeśli ma to zastosowanie. |
| Paragraphs | int? | Całkowita liczba akapitów znalezionych w dokumencie, jeśli ma to zastosowanie. |
| Pages | int? | Całkowita liczba stron dokumentu, jeśli ma to zastosowanie. |
| Slides | int? | Całkowita liczba slajdów w dokumencie prezentacji. |
| HiddenSlides | int? | Liczba ukrytych slajdów w dokumencie prezentacji. |
| MultimediaClips | int? | Całkowita liczba klipów dźwiękowych lub wideo, które są obecne w dokumencie. |
| Notes | int? | Liczba slajdów w prezentacji zawierającej notatki. |
| **Extra Document Properties** | | |
| DocimentId | int? | Identyfikator dokumentu nadawany w wersji Word 2010. |
| PersistentDocimentId | Guid? | Unikatowy identyfikator dokumentu nadawany w wersji Word 2013. |
| ConflictMode | bool | Występują sprzeczne zmiany przy zapisywaniu dokumentu |
| Rsids | RsIds | Lista wszystkich wartości identyfikatora zapisu wersji |

Interfejs ICustomDocumentProperties reprezentuje kolekcję właściwości dokumentu definiowanych przez użytkownika. Właściwości definiowane przez użytkownika są reprezentowane przez interfejs ICustomDocumentProperty, który dodaje do interfejsu IDocumentProperty trzy dodatkowe właściwości:

* FormatID: Guid? – określa format właściwości. domyślnie ma wartość {D5CDD505-2E9C-101B-9397-08002B2CF9AE}.
* PropertyID: int? – unikatowo wiąże właściwość niestandardową z właściwością OLE. Domyślnie ma wartość 2.
* LinkTarget: string? – określa nazwę zakładki w bieżącym dokumencie, z której należy wyodrębnić wartość tej niestandardowej właściwości dokumentu.

Wśród dodatkowych właściwości znajduje się obiekt klasy RsIds, który jest kolekcją identyfikatorów „rewizji”, czyli kolejnych zapisów dokumentu.

## Motyw

## Definicje stylów

W projekcie modelu dokumentu jest interfejs/klasa Style oraz kolekcja stylów Styles.

Kolekcja Styles umożliwia dostęp wyliczeniowy do wszystkich stylów oraz dostęp indeksowany po numerze stylu, po nazwie oryginalnej (angielskiej nazwie stylu wbudowanego), po nazwie lokalnej i po identyfikatorze stylu. Kolekcja stylów składa się z dwóch list – listy stylów wbudowanych i listy stylów użytkownika.

Bez dostępu do aplikacji Worda nie ma dostępu do definicji wszystkich stylów wbudowanych. Lista stylów wbudowanych jest inicjowana na podstawie informacji wcześniej pozyskanych z polskiej wersji Worda. Informacje te zawierają typy stylów oraz nazwy oryginalne (angielskie) i lokalne (w języku polskim), ale nic poza tym.