

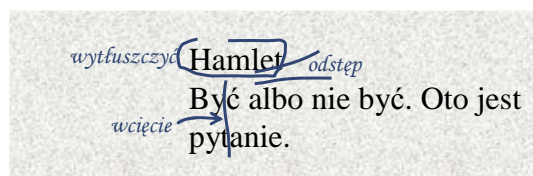
Historia rozwoju technik znakowania tekstu

Znakowanie tekstu

Markup:

*the process of marking manuscript copy for typesetting with
directions for use of type fonts and sizes, spacing, indentation, etc.*

The Chicago Manual Of Style



Znakowanie tekstu w epoce komputerów

Treść

Hamlet Być albo nie być. Oto jest pytanie

+

Formatowanie, adjustacja

{nowy_wiersz} {bold} {wyłącz_bold} {wcięcie}

=

Dokument

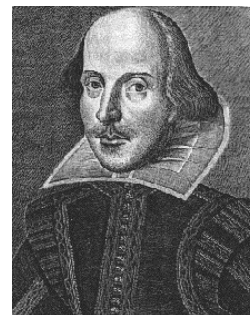
Hamlet

Być albo nie być. Oto jest pytanie.



Przykłady języków znakowania

- **Frame (MIF)** `<Font <FTag `B`>>`
`<String `Hamlet`>`
- **QuarkXPress** `Hamlet`
- **RTF** `{\b\f5\cf1 Hamlet}`
- **Ventura** `Hamlet<D>`
- **TeX/LaTeX** `\textbf{Hamlet}`
- **PostScript** `/Times-BoldR 900 ff`
`(Hamlet)W`
- **HTML** `Hamlet`



Rozwój języków uogólnionego znakowania tekstu

- 1969: GML – Generalized Markup Language (IBM; Goldfarb, Mosher, Laurie).
- 1986: SGML – Standard Generalized Markup Language, ISO 8879:1986.
- 1991: powstaje World Wide Web.
- 1994: HTML 2.0 zdefiniowany jako zastosowanie SGML-a.
- 1998: XML – Extensible Markup Language, World Wide Web Consortium.

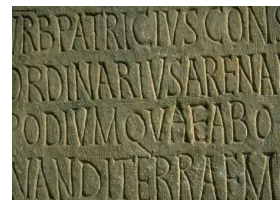


2008-10-02 Historia rozwoju technik znakowania tekstu

5

Korzenie

- Lata 60-te XX wieku:
 - 1967 – William Tunnicliffe, prezes Graphic Communications Association, podczas spotkania w Canadian Government Printing Office przedstawia ideę oddzielenia zawartości informacyjnej dokumentów od ich formatu,
 - Stanley Rice proponuje użycie uniwersalnych znaczników do znakowania struktury tekstu,
 - projekt GenCode definiuje sposób oznaczania tekstu ukierunkowany na jego strukturę.



2008-10-02 Historia rozwoju technik znakowania tekstu

6

Korzenie: INTIME

- INTIME – Interactive Textual Information Management Experiment:
 - projekt badawczy Charlesa Goldfarba (IBM Cambridge Scientific Center, koniec lat 60-tych XX wieku),
 - prototyp zintegrowanego systemu przetwarzania tekstu:
 - edycja tekstu,
 - repozytorium dokumentów,
 - wyszukiwanie;
 - wykorzystane technologie:
 - „maszyny wirtualne” na mainframie IBM 360,
 - *concurrent access to a disk file*,
 - *context editors*.



Edytor kontekstowy

```
LOCATE /researchers/  
researchers. A system which integrates  
CHANGE /researchers/analysts/  
analysts. A system which integrates  
CHANGE /edit/edit/ *  
In online systems, text editing is  
are known as "context" editors. They  
NEXT  
provide a retrieval capability: e.g.,  
QUIT
```

Wnioski z projektu INTIME

- Wyszukiwanie jest efektywniejsze gdy znana jest struktura i przeznaczenie poszczególnych fragmentów tekstu.
- Opracowano heurystykę odgadującą strukturę tekstu, ale zauważono potrzebę oznaczania struktury w dokumencie źródłowym.
- Istniejące (wówczas) języki znakowania tekstu koncentrują się na wyglądzie, a nie strukturze czy znaczeniu tekstu.

Na podst.: C. Goldfarb, *SGML: The Reason Why and the First Published Hint*, Journal of the American Society for Information Science, Volume 48, Number 7 (July 1997)

GML i SGML

- GML:
 - 1969, Charles Goldfarb, Edward Mosher, Raymond Lorie,
 - powstał jako język makr do edytora IBM SCRIPT:
 - opisujących strukturę dokumentu,
 - zamienianych na znaczniki formatujące.
 - możliwe było rozszerzanie początkowego zbioru znaczników.
 - narzędzie pozwalało na definiowanie wielu „profilu” wizualizujących dokument.
- SGML:
 - pierwsze wersje robocze w 1980.
 - standard ISO w 1986.
 - rozwinięty potomek GML.

Wokół SGML-a

- Pierwsze szerzej znane zastosowania SGML-a:
 - Electronic Manuscript Project, Association of American Publishers, 1987,
 - CALS – Computer-Aided Acquisition and Logistic Support, US Department of Defense, MIL-M-28001, February 1988.
- Standardy pokrewne:
 - DSSSL – Document Style Semantics and Specification Language,
 - HyTime:
 - meta-notacja dla linków,
 - opis struktur multimedialnych, rozciągniętych w czasie.

World Wide Web Consortium (W3C)

- Kuźnia standardów internetowych, np.:
 - HTML – Hyper Text Markup Language,
 - HTTP – Hyper Text Transfer Protocol,
 - CSS – Cascading StyleSheets,
 - ...
- XML – Extensible Markup Language:
 - najważniejsza rekomendacja ostatnich lat,
 - twórcy: Tim Bray (Netscape), Jean Paoli (Microsoft), C.M. Sperberg-McQueen (University of Illinois).
- Obecne dominują prace nad standardami związanymi z XML-em.

Programy i ich formaty

- Prawie każda aplikacja wprowadza swój wewnętrzny format.
- Nowe wersje tej samej aplikacji wprowadzają zmiany do używanego formatu:
 - wsteczna kompatybilność,
 - brak możliwości zapisu do formatu poprzednich wersji.
- Aplikacje dostarczają konwerterów:
 - tylko do najpopularniejszych formatów,
 - możliwość utraty danych podczas konwersji.



Standardy

- Nie istnieją uznane standardy.
- Istnieją substandardy w różnych dziedzinach:
 - dokumenty biurowe: Microsoft Word,
 - teksty naukowe: Postscript, TeX,
 - Internet: HTML, GIF, JPG,
 - elektroniczna wymiana danych: EDIFACT.
- Standard musi być:
 - własnością publiczną,
 - otwarty i jawny,
 - niezależny od konkretnego producenta oprogramowania.

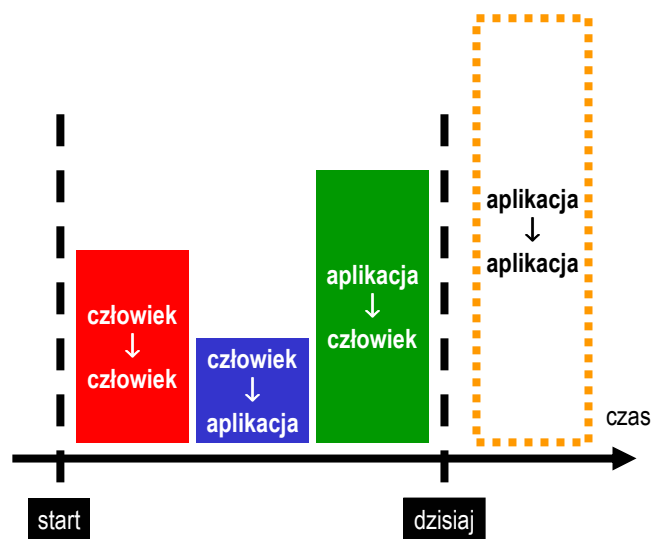


Potrzeba struktury

- Masa informacji cyfrowej powoduje potrzebę struktury:
 - jeden format dokumentu nie wystarczy dla 5 miliardów ludzi,
 - ale nie możemy operować milionami niekompatybilnych formatów.



Ewolucja Internetu



Idea SGML/XML (1)

Oddzielenie znaczenia tekstu od sposobu prezentacji

<OSOBA MÓWIĄCA>Hamlet</OSOBA MÓWIĄCA>
<WYPOWIEDŹ>Być albo nie być.
Oto jest pytanie.</WYPOWIEDŹ>



Sposób prezentacji

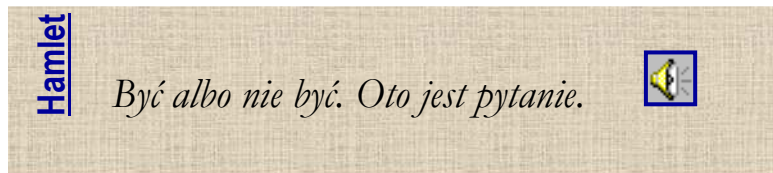
- OSOBA MÓWIĄCA
 - nowy akapit
 - do lewej
 - wytłuszczenie
- WYPOWIEDŹ
 - nowy akapit
 - wcięcie na 2 cm
 - do lewej

Hamlet

Być albo nie być. Oto jest pytanie.

Inny sposób prezentacji

- OSOBA MÓWIĄCA
 - na marginesie
 - tekst pionowo
 - niebieski
 - hiperlink do opisu postaci na początku dramatu
- WYPOWIEDŹ
 - nowy akapit
 - kursywa
 - ew. użyj syntezy mowy z ustawieniami dla OSOBY MÓWIĄCEJ



Idea SGML/XML (2)

Stworzenie najodpowiedniejszego modelu dla naszych własnych dokumentów.

```
<OSOBA MÓWIĄCA>Hamlet</OSOBA MÓWIĄCA>  
<WYPOWIEDŹ> <NUDA> Być albo nie być.  
Oto jest pytanie.</NUDA> </WYPOWIEDŹ>
```

Najodpowiedniejszy model

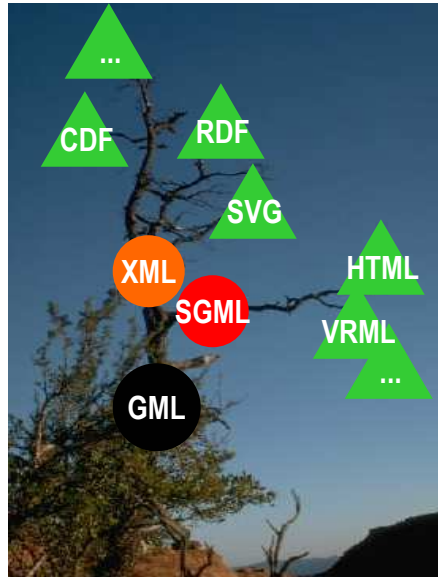
- Przykłady:
 - encyklopedia: <nazwisko>, <imie>, <ur>, <zm>, <wymowa>, <etymologia>, <liczba-mieszk>
 - prawo: <promulgator>, <rocznik>, <poz>, <art> <sąd>, <sygn-wyroku>, <teza>
 - dokument techniczny: <part-number>, <function-name>
 - patenty: <wynalazca>, <nr-zgłoszenia>
 - ubezpieczenia: <data-polisy>, <wartość-polisy>

Język – metajęzyk

- Stan wyjściowy:
 - Wieża Babel (brak wspólnego języka),
 - czy w ogóle możliwy jeden wspólny język?
- Wspólny metajęzyk:
 - znana gramatyka,
 - jednolita metodologia,
 - takie same narzędzia.
- Dowolnie wiele języków specyficznych dla zastosowań.



Genealogia XML-a



2008-10-02

Historia rozwoju technik znakowania tekstu

23

Co to jest XML?

- XML to nie język programowania.
- XML to sposób zapamiętywania danych wraz z ich strukturą w dokumencie tekstowym:
 - otwarty,
 - elastyczny,
 - bezpłatny,
 - niezależny od platformy sprzętowej.
- XML to rama składniowa do tworzenia języków specyficznych dla zastosowań.
- Użycie XML-a nie zwalnia od myślenia (analizy, projektowania, ...)

2008-10-02

Historia rozwoju technik znakowania tekstu

24

Jak wygląda XML?

```
<?xml version="1.0"?>
<zeznanie-sprawcy nr="1313/2001">
  <autor>st. asp. Jan Łapówka</autor>
  <miejsce>Dołowice Górne</miejsce>
  <treść>Wypadek dnia
  <data>13.10.2001r</data>
  o godzinie <godzina>13:13</godzina>
  (<dzien-tygodnia>piątek
  </dzien-tygodnia>) miał miejsce nie
  z mojej winy. <poszkodowany>Alojzy
  M.</poszkodowany> nie miał żadnego
  pomysłu w którą stronę uciekać, więc
  go przejechałem.</treść>
</zeznanie-sprawcy>
```

Deklaracja XML
Element główny
Atrybut
Element
Znacznik początkowy
Znacznik końcowy
Zawartość tekstowa

HTML ↔ XML

- Znaczenie elementów i ich atrybutów z góry określone.
- Interpretację elementów określa standard, a w praktyce przeglądarki internetowe.
- To, co jest poprawne również określają przeglądarki internetowe.
- Znaczenie elementów i ich atrybutów określa użytkownik lub aplikacja.
- `<p>` może w jednym dokumencie oznaczać **paragraf**, w drugim **pomoc**, a w trzecim **pismo odręczne**.
- Poprawność XML-a jest ściśle określona przez specyfikację.

SGML ↔ XML

- Filozofia: jeden duży system zarządzania treścią.
- Konieczność definiowania struktury.
- Skomplikowana składnia, wiele opcji.
- Trudność tworzenia parserów.
- Bardzo drogie narzędzia.
- Filozofia: wiele małych komunikujących się ze sobą modułów.
- Opcjonalne definiowanie struktury.
- Uproszczona składnia.
- Łatwość tworzenia parserów.
- Darmowe narzędzia.

Klasy zastosowań XML-a

Zarządzanie dokumentami, treścią, wiedzą:

- Pierwotne zastosowanie SGML-a.
- Dokumenty tworzone przez człowieka i przeznaczone dla człowieka.
- Długi czas życia dokumentów.
- Typowy model mieszany zawartości.

Elektroniczna wymiana danych, integracja aplikacji:

- Nowa klasa zastosowań XML-a.
- Dokumenty tworzone oraz przetwarzane automatycznie.
- Dokumenty tworzone tylko na czas komunikacji.
- Konieczność dokładnego kontrolowania struktury i zawartości.

Dwie twarze XML-a

Dokument tekstowy:

```
<zeznanie-sprawcy>
Wypadek dnia <data>
13.01.2001 r.</data>
o godzinie <godzina>13.13
</godzina> (<dzien-
tygodnia>piątek
</dzien-tygodnia>) miał
miejsce nie z mojej winy.
<poszkodowany>Alojzy
M.</poszkodowany> nie miał
żadnego pomysłu w którą
stronę uciekać, więc go
przejechałem.
</zeznanie-sprawcy>
```

Baza danych:

```
<zamowienie>
  <pozycja>
    <nazwa>Papier</nazwa>
    <jednostka>ryza
  </jednostka>
    <ilosc>3</ilosc>
  </pozycja>
  <zamawiajacy id="123456">
    <imie>Szymon</imie>
    <nazwisko>Zioło
  </nazwisko>
    <firma>ABG Ster-Projekt
  </firma>
  </zamawiajacy>
</zamowienie>
```

Literatura: historia XML-a

- Charles F. Goldfarb's SGML Source Home Page:
🌐 www.sgmlsource.com
- Wypych, W., *Na początku był rękopis, czyli o historii XML-a:*
📅 Software 2.0, 6/2001