

# e- Przypisy – miejsce i przyszłość przypisów w dobie cyfrowej

Wydawnictwo Ruthenus, kontakt: [bogdan.szczurek@ruthenus.pl](mailto:bogdan.szczurek@ruthenus.pl)

Zagadnienie e-przypisu, jako przypisu przeszczepionego z tradycyjnych mediów do mediów cyfrowych, przypomina nieco rzeczne rozlewisko, które, choć stanowi niezaprzeczalną całość, składa się z rozłącznych, a jednak przeplatających się niekiedy ze sobą nurtów. Dopóki nie zbadamy ich wszystkich, nie będziemy w stanie uzyskać choćby przybliżonego obrazu „starego” przypisu w jego współczesnej inkarnacji. Dlatego, uzbrojeni w konieczną wszystkim prawdziwym odkrywcom ciekawość, wyruszymy teraz na wyprawę po „delcie »Przypisu«” ku mającej na horyzoncie przyszłości.

## Przede wszystkim wygląd

Kiedy przyjrzymy się tradycyjnym, prekomputerowym, narzędziom typograficznym, bez większego trudu zauważymy pewną prawidłowość. Niezależnie od tego, czy weźmiemy pod uwagę skład ręczny, w którym zecer pieczołowicie ustawiał w odpowiedniej kolejności czcionki, czy pierwsze generacje składu maszynowego w znacznym stopniu przyspieszające i ułatwiające tę czynność, dojdziemy do wniosku, że narzędzia i metody pracy są podporządkowane jednemu celowi: finalnemu wy-

gładowi druku. Na pierwszy rzut oka taka obserwacja może wydawać się nieistotna i bliska truizmowi, jednak w istocie nie jest to wcale fakt trywialny, szczególnie jeśli rozważymy jego konsekwencje w odniesieniu do myślenia o tekście jako nośniku informacji.

Tysiące lat rozwoju najpierw piśmiennictwa, a potem drukarstwa, zwłaszcza w tradycji zachodniej, doprowadziły do ewolucyjnego wytworzenia pewnych wizualnych struktur nierozzerwalnie i jednoznacznie, choć jednak nieformalnie, związanych ze strukturą informacji zawartej w tekście. Są to takie struktury, jak choćby odstęp międzywyrazowy (popularna „spacja”), akapit, tytuł książki, tytuł rozdziału czy wreszcie przypis stanowiący główny temat prezentowanych rozważań. Zadomowiły się one tak bardzo w zbiorowej świadomości, że najczęściej odbieramy je i interpretujemy zupełnie bezrefleksyjnie, nawet wtedy, gdy ich forma wizualna podlega sporym wariacjom w poszczególnych dziełach. Ich znajomość jest dla nas naturalna i pozwala nam m.in. na odebranie i odtworzenie logicznej struktury informacji, nadanej tekstowi przez autora czy też redaktora. Taka struktura, np. struktura rozdziałów, jest w dużej mierze niezależna od treści konkretnego tekstu i dlatego możemy na rozdziały dzielić zarówno wielkie dzieła prozy, jak i rozprawy naukowe czy instrukcje obsługi. Istotne jednak jest to, że struktura ta daje nam możliwość arbitralnego podziału informacji na mniejsze fragmenty, co ułatwia naszemu umysłowi przyjęcie informacji i działanie w odniesieniu do niej. Można powiedzieć, że w ten sposób umysł otrzymuje informację „wstępnie przetrawioną”. Struktury reprezentowane wizualnie dają nam też możliwość komentowania czy uzupełniania głównej treści bez szkody dla płynności jej odbioru (przypis, marginalia), możliwość łatwiejszego oglądu i porównania faktów (tabele) czy stosunkowo szybkiego dotarcia do interesujących nas informacji (spisy i indeksy). Nie jest to bynajmniej pełny spis wszystkich struktur typograficzno-logicznych, jednak reprezentuje on wystarczająco ich spektrum użytkowe i funkcjonalne.

Do czasu, kiedy człowiek zaczął używać komputerów w celu gromadzenia i przetwarzania informacji, tekst pozostawał na papierze, struktura tekstu miała swój wyraz tylko w jego wyglądzie i była wykorzystywana wyłącznie przez człowieka jako twórcę bądź odbiorcę tego tekstu. Z tego względu niemal każdy krój pisma, na który się natkniemy, będzie miał nazwę opisującą przede wszystkim jego wygląd, a nie logiczny komponent tekstu, gdzie miałby on znaleźć zastosowanie. Owszem, okazjonalnie znajdziemy krój zaprojektowany z przeznaczeniem „do składu tytułów”, ale nie uda nam się natrafić na krój do składania przypisów czy tekstu tabel.

Znajdziemy „Caslony”, „Helvetiki” czy „Timesy”, znajdziemy ich odmiany italikowe, pochyle bądź grube, wreszcie znajdziemy ich poszczególne rozmiary wyrażone np. w punktach. Na próżno jednak szukać będziemy w książkach z próbkami tradycyjnych czcionek<sup>1</sup> lub też na klawiaturach linotypu czy monotypu<sup>2</sup> znaków opracowanych i przeznaczonych specjalnie do składu przypisów<sup>3</sup>.

Dopiero w tym świetle można właściwie zrozumieć tytułowe „przede wszystkim wygląd” w odniesieniu do narzędzi i metod, których jedynym celem było „zabarwienie papieru w odpowiednim miejscu”. Narzędzia i metody nie niosły bowiem żadnej bezpośredniej informacji o strukturze tekstu będącego przedmiotem ich działania. Jedynie ludzie zaangażowani w proces używania tych narzędzi mieli świadomość logicznej budowy tekstu i dbali o to, żeby ta budowa znalazła swoje odzwierciedlenie w układzie manuskryptu czy druku. Można powiedzieć, że struktura była zachowywana i przenoszona (i jest w dalszym ciągu) dzięki procesowi, który można byłoby określić mianem enkapsulacji wizualnej – „tłumaczeniem” struktury logicznej tekstu na serię form i konwencji graficznych. Jedną z takich form i konwencji pozwala nam również na rozpoznawanie obiektu, który określamy mianem przypisu.

Zanim przejdziemy dalej, warto jeszcze zauważyć, że tekst pojmowany tradycyjnie, czyli tekst pisany bądź drukowany, nie niesie ze sobą niczego, co byłoby w jakikolwiek sposób ukryte przed jego odbiorcą – „początkiem” i „końcem” takiego tekstu jest wyłącznie sfera wizualna. Znów stwierdzenie to może wydawać się pozornie banalne, jednak ciąg dalszy artykułu powinien to wrażenie skutecznie zatrzeć.

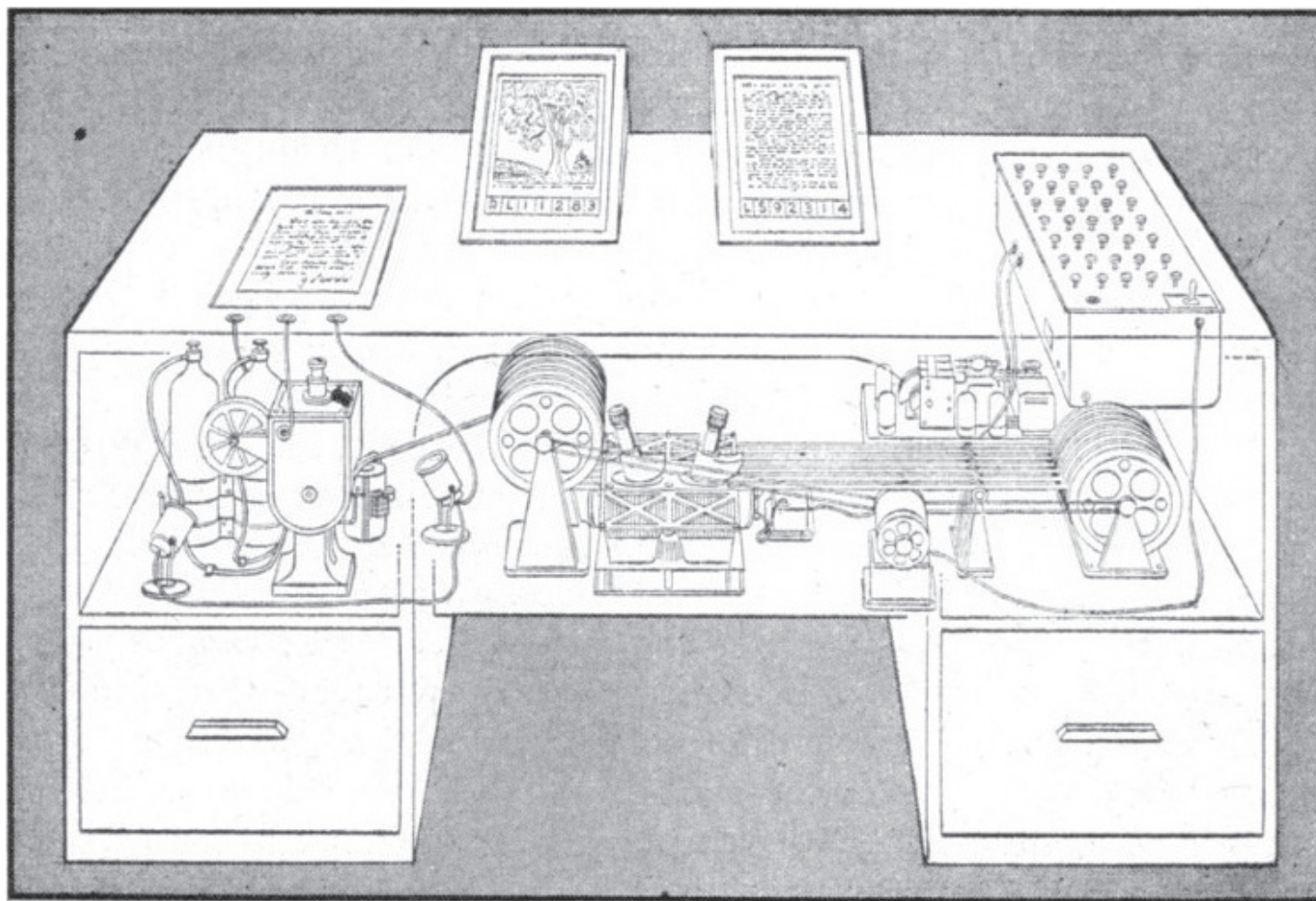
## Przede wszystkim struktura

Bardzo trudno określić, kiedy dokładnie ludzie zaczęli interesować się tekstem samym w sobie, pod kątem jego logicznej organizacji, jej elementów składowych i zależności między nimi. Musiało to wszakże nastąpić na bardzo wczesnym etapie rozwoju piśmiennictwa, skoro nawet w wielowiekowych manuskryptach można odnaleźć sugestie bądź bezpośrednie użycie np. struktur rozdziałowych. Jednak prawdziwy przełom w patrzeniu na tekst jako strukturę nastąpił dopiero wraz z nastaniem świtu współczesnych technik informacyjnych, a w szczególności komputerowych, i choć do tej pory rozważaliśmy nurt tradycyjnej typografii, geneza przełomu, o którym mowa, zdaje się mieć korzenie w tradycji bibliotekarskiej, a nie drukarskiej.

Bardzo znaczącym śladem wspierającym to stwierdzenie jest artykuł Vannevara Busha<sup>4</sup> opublikowany w amerykańskim piśmie „The Atlantic” w 1945 roku, noszący tytuł *Jak możemy myśleć*<sup>5</sup>. Artykuł ten jest zbiorem refleksji autora na temat m.in. sposobu działania ludzkiego intelektu, które prowadzą do opisu koncepcji maszyny wspierającej ten intelekt i mającej także potencjał rozszerzenia jego możliwości. Maszynie, nigdy zresztą w formie opisanej w artykule niezrealizowanej w praktyce, autor nadał nazwę **memex**<sup>6</sup>. Miała mieć ona wygląd zwykłego biurka (il. 1), w którym zainstalowano by **przezroczyste ekrany**, na których byłyby wyświetlane fotografie stron dokumentów i książek przechowywane na mikrofilmach. Ten obraz, jak do tej pory, nie różni się od zwykłego bibliotecznego stanowiska do pracy z mikrofilmami, jednak tym, co miało stanowić o wyjątkowości **memexu**, była jego daleko idąca automatyzacja. Użytkownik takiej maszyny nie tylko miał mieć bowiem dostęp do całej biblioteki mi-

krofilmów, którą był w stanie modyfikować według potrzeb i uznania, ale mógł również szybko odwoływać się do określonych miejsc tej biblioteki za pomocą **kodów** będących czymś na pograniczu indeksu i katalogu bibliotecznego.

Idea **memexu** nie ogranicza się jednak do w pełni zautomatyzowanej biblioteki zamkniętej w pojedynczym biurku z maszyną w roli bibliotekarza. W dalszej części artykułu autor pisze o jeszcze jednej właściwości swojego automatu, mianowicie o możliwości tworzenia i zachowywania powiązań definiowanych przez operatora **memexu** między treściami zgromadzonymi w tej specyficznej bazie wiedzy. Załóżmy przykładowo, że osoba korzystająca z **memexu** pisze pracę na temat łuków bojowych. Swoje badania może rozpocząć od wyszukania odpowiedniego hasła w encyklopedii, które stanowi źródło tyle wiarygodnych, co i skąpych wiadomości, więc jest tylko wstępem do dalszych poszukiwań. W ich toku operator maszyny znajduje np. ryciny, zdjęcia czy schematy



Il. 1. Ilustracja z przedruku eseju Vannevara Busha w magazynie „Life” z 10 września 1945 roku

łuków, znajduje opracowania historyczne i techniczne, wreszcie może sam dodawać do puli informacji własne notatki. Wszystkie te elementy równocześnie łączy ze sobą w coś na kształt ścieżki własnych myśli, która jest zachowywana bezterminowo. W ten sposób nawet po wielu latach można dokładnie odtworzyć i prześledzić, skąd wychodziły i jak przebiegały poszukiwania informacji. Co więcej, wszystkie istotne elementy całościowego obrazu wiedzy nabytej przez człowieka są ze sobą połączone, nawet jeśli znajdują się w różnych dokumentach i książkach – strona książki z ilustracją w innej książce, odręcznie narysowany diagram z hasłem encyklopedii.

Vannevar Bush opisuje swój pomysł i obserwacje bardziej szczegółowo i wspomina jeszcze o kilku dodatkowych cechach **memexu**, my jednak poprzestaniemy na tym.

Nie trzeba wyjątkowej spostrzegawczości, żeby zauważyć ludzkie podobieństwa między **memexem** a pracą ze współczesną przeglądarką internetową, choć, oddając sprawiedliwość temu pierwszemu, pewne dobre rozwiązania w nim zaproponowane nadal nie doczekały się ziszczenia. Niemniej jednak każda idea jest sumą swoich inspiracji skomponowanych w całość przez ludzki umysł, dlatego nie pozostaje bezpodstawną nadzieją, że dawne, intrygujące pomysły przyniosą jednak owoc godny marzeń ich twórców, jeśli tylko pomysły te staną się czyjąś inspiracją.

Ten opis koncepcji maszyny sprzed lat, choć obszerny, jest jednak w dwójnasób konieczny. Po pierwsze, do wykazania „bibliotecznych” źródeł zainteresowań strukturą tekstu, a po drugie, do zwrócenia uwagi czytelnika na pewien istotny, lecz nieoczywisty fakt. Mówiąc o automatycznym dostępie do informacji i maszynie przejmującej kompetencje bibliotekarza, musimy zdać sobie sprawę z tego, że dowolna maszyna jest niczym innym jak bezwolnym i bezrefleksyjnym wykonawcą rozkazów. Przy tym te rozkazy muszą być konieczne jednoznaczne, bo ich odbiorcą nie jest intelekt mający zdolność interpretacji. To z kolei oznacza, że aby **automatyczny bibliotekarz** mógł w ogóle cokolwiek odnaleźć w powierzonym jego pieczy magazynie wiedzy i przedstawić wynik poszukiwań człowiekowi, niezbędny jest jednoznaczny mechanizm identyfikacji informacji czytelny dla maszyny. W przypadku **memexu** miały to być specjalne symbole naniesione w ustalonych miejscach mikrofilmów. Jest to bezpośrednia inspiracja istniejącymi konwencjami i standardami bibliotecznej organizacji i klasyfikacji zbiorów, z tą różnicą, że informacje kategoryzacyjne są przeznaczone nie dla człowieka-bibliotekarza, lecz dla automatu.

Rozwinięciem myśli Vannevara Busha są pojęcia **hipertekstu** i **hipermediów** ukute przez Theodora „Teda”

Nelsona<sup>7</sup>. Pierwsze z nich jest dzisiaj w powszechnym użyciu przede wszystkim za sprawą sukcesu pomysłów i prac Tima Bernersa-Lee<sup>8</sup>, z których w ich dojrzałej już formie korzystamy na co dzień za pośrednictwem przeglądarek internetowych. Trzeba jednak zaznaczyć, że **hipertekst**, z którym obcujemy za ich pomocą, to tylko niedoskonała wersja tego postulowanego przez Nelsona i opisywanego, choć nie wprost, przez Busha.

W ten sposób nakreśliliśmy początki kolejnego nurtu ukazującego tekst i materiały graficzne nie od strony ich wyglądu, ale logicznej budowy. Co więcej, wprowadza on *explicite* do tych materiałów dodatkowe oznaczenia, niekoniecznie przeznaczone dla człowieka, powtarzające, wzbogacające i rozszerzające ich podstawowe struktury wynikające z interpretacji ich kompozycji graficznej.

## Programowanie książki

W tym miejscu nasza historia nabiera tempa, przeskakujemy bowiem do połowy XX wieku, kiedy zaczęły pojawiać się pierwsze maszyny fotoskładowe. Stanowiły one drugą generację maszyn składających, zaraz po maszynach, takich jak linotyp czy monotyp, których wynikiem działania były odlane z miękkich stopów kompletne wiersze tekstu. Maszyny fotoskładowe wytwarzały już nie metalowe odlewy, ale całe kolumny tekstu naświetlone na specjalnym światłoczułym podłożu. Metalowe matryce maszyn pierwszej generacji zastąpiły dyski, taśmy, płyty lub innego kształtu i formy przezroczka, na których były zarejestrowane obrazy glifów czcionek. Pozwoliło to na zmniejszenie rozmiarów maszyn, łatwiejszą wymianę rodzin czcionek, ich większe upakowanie (można było bez wymiany nośnika korzystać z większej liczby znaków), a także umożliwiło w pewnych modelach fotoskładarek skalowanie glifów, co było sporym osiągnięciem i udogodnieniem. Co więcej, sterowanie maszyn do fotoskładu za pomocą taśmy perforowanej, na której były zapisane wszystkie parametry prezentacyjne tekstu, nie należało wcale do rzadkości. Choć w dalszym ciągu maszyny te stanowiły granicę, poza którą istotny był tylko wygląd tekstu, granica ta stała się łatwiejsza do pokonania.

Następny przeskok, jakiego dokonamy, to przeskok do połowy lat sześćdziesiątych XX wieku, kiedy to komputery, nieustannie rozwijane od czasów drugiej wojny światowej, z pierwotnego narzędzia mariażu wojska i nauki stały się technologią stosunkowo okrzepłą i dostępną. Ze wzrostem

dostępności zaczęły też mnożyć się pomysły ich potencjalnego zastosowania i wymogi ich użytkowników. Jedną z nich było formatowanie tekstu na potrzeby komputerowego terminala. W tamtych czasach funkcję terminali pełniły... zwykle telexy. Za ich pomocą użytkownik mógł wysyłać wiadomości do komputera, a komputer mógł na te wiadomości odpowiadać. Mimo że telex to w zasadzie maszyna do pisania, którą można sterować zdalnie, więc i jego możliwości graficzne nie były mocno rozwinięte, istniała potrzeba stworzenia programu, który byłby w stanie uwolnić człowieka od konieczności ręcznego formatowania tekstu. Czytelnicy, którzy korzystali lub korzystają z maszyny do pisania, z pewnością wiedzą, jak uciążliwe może być umieszczanie napisów na środku strony albo utworzenie jednolitego odstępu od lewego marginesu na całej wysokości kolumny. Te same problemy mieli użytkownicy pierwszych terminali. Dlatego powstał program o nazwie RUNOFF. Komplementarnie do niego napisano również program TYPSET, który można zasadniczo uznać za edytor tekstu. Jeśli dokument tekstowy zawierał odpowiednie instrukcje przeznaczone dla programu RUNOFF, to ten był w stanie przetworzyć taki tekst, wymieszany z instrukcjami, na odpowiednio wyglądający ciąg znaków wypisywany następnie przez terminal. Przykładowo: użytkownik umieszczał w teście fragment

```
.CENTER
Centered Title,
```

a RUNOFF dbał już o to, żeby przed tekstem „Centered Title” pojawiła się odpowiednia liczba spacji tak, żeby napis ten znalazł się na środku strony.

Rozwój i zmiany w technice komputerowej spowodowały, że na początku lat siedemdziesiątych XX wieku konieczne stało się utworzenie nowej inkarnacji RUNOFF-a, którą nazwano krócej **roff** i wykorzystywano np. do formatowania stron dokumentacji systemu UNIX. Kilka lat później, w 1973 roku, laboratoria firmy Bell, gdzie powstał **roff**, zakupiły dla swoich potrzeb maszynę fotoskładową GSI C/A/T. Był to nie lada wysiłek finansowy, szczególnie w tamtym czasie, i niewiele instytucji mogło sobie na taki zakup pozwolić. Maszyna była sterowana za pomocą taśmy perforowanej, jednak dostosowanie jej do bezpośredniego sterowania przez komputer było stosunkowo proste. Mając do dyspozycji takie narzędzie jak maszyna fotoskładowa, jeszcze tego samego roku Joseph F. Ossanna, wychodząc z oryginalnej wersji **roffa** jako

Nie uda nam się  
natrafić na krój  
do składania przypisów  
czy tekstu tabel

bazy, napisał dwa nowe programy: **nroff** i **troff**<sup>9</sup>. Ten drugi był szczególnie interesujący, ponieważ dawał swojemu użytkownikowi dostęp do wysokiej jakości typografii oferowanej przez maszynę C/A/T. Dość powiedzieć, że po raz pierwszy w historii komputer umożliwiał wybór kroju pisma czy wykorzystanie spacji o różnych szerokościach. **Nroff** z kolei był w zasadzie **troffem**, który jednak miał obsługiwać proste drukarki i terminale, dlatego był on tak skonstruowany, żeby ignorować zaawansowane instrukcje typograficzne, których i tak nie był w stanie finalnie oddać.

**Troff** jako program ewoluował. Początkowo jego możliwości niewiele różniły się od możliwości jego przodka – podobnie jak **roff** był w stanie samodzielnie dzielić otrzymany tekst na strony, numerować te strony i dodawać do nich nagłówki – z czasem jednak zakres zadań typograficznych, które mógł wypełnić, zwiększał się. Była to zasługa przede wszystkim Briana Kerninghana, który przejął opiekę nad **troffem**<sup>10</sup>. Pod koniec lat siedemdziesiątych XX wieku okazało

się, że system C/A/T wymaga wymiany na nowszy. Wobec tego zdecydowano się na kupno maszyny Mergenthaler Linotron 202. Linotron 202 był już reprezentantem trzeciej generacji maszyn. W odróżnieniu od maszyn wcześniejszych, maszyna ta przechowywała kroje pisma

nie na przezroczach, ale w pamięci wbudowanego komputera<sup>11</sup>. Ponadto obraz był naświetlany na materiale światłoczułym nie przy pomocy skomplikowanych zestawów soczewek, ale z wykorzystaniem niewielkiego, za to bardzo dokładnego, ekranu CRT<sup>12</sup>. Dawało to olbrzymi potencjał, bo oznaczało, że oto pojawiło się narzędzie sterowane przez komputer, które jest w stanie oddać na papierze niemal dowolny kształt. Niestety producent nie zgodził się na uchylenie rąbka tajemnicy osnuwającej sposób przechowywania kształtów czcionek na dyskach (zapewne z obawy przed konkurencją). Wobec tego kilku pracowników laboratoriów Bell postanowiło samodzielnie rozwikłać tę zagadkę. Ich starania zakończyły się sukcesem<sup>13</sup> i były bodaj pierwszym przykładem tzw. inżynierii odwrotnej, czyli poznania istotnych mechanizmów działania urządzenia na podstawie badania jego sprawnego egzemplarza. Sukces ten pozwolił na wykorzystanie maszyny w sposób wykraczający nawet poza wyobrażenia jej producenta, udało się bowiem przy jej pomocy wykonywać nawet prostą grafikę. Był to bardzo ważny impuls, który sprawił, że **troff** zmienił się z programu przeznaczonego do zarządzania tylko jedną, konkretną maszyną do fotoskładu, w program dający spore możliwości graficzne i typograficzne oraz program,

który był w stanie obsługiwać różne maszyny, a potem nawet generować różne formaty plików jak PostScript czy PDF. Był w stanie tworzyć tabele, rysować schematy, składać równania matematyczne i wreszcie pozwalał osobie piszącej na łatwe dodawanie przypisów do tekstu, które to przypisy mógł automatycznie numerować i umieszczać w odpowiednim miejscu strony. To ostatnie stało się możliwe dzięki tzw. **makrom**, czyli specyficznym programom napisanym z wykorzystaniem instrukcji danego systemu. Jeśli okazuje się, że w czasie pracy z programem użytkownik często powtarza wciąż te same operacje, to może je połączyć w **makro** i w ten sposób stworzyć nową operację powodującą automatyczne wykonanie wszystkich innych w niej zawartych. Ta funkcjonalność **troffa** pozwala nam ograniczyć operację wstawiania przypisu do umieszczenia w dokumencie dosłownie kilku instrukcji. Przykładowo, poddając działaniu **troffa** tekst:

```
>Lorem ipsum dolor sit amet.\|.\\.*  
.FS  
Consectetur adipiscing elit.  
.FE,
```

osiągniemy wynik widoczny na ilustracji 2. Znaki `\\.*` oznaczają, że wygenerowany przypis ma mieć numerację arabską, instrukcje `.FS` i `.FE` oznaczają zaś odpowiednio początek i koniec tekstu przypisu<sup>14</sup>. Wagi tego, wydawać by się mogło, prostego faktu nie można jednak przecenić, bo oto wskazujemy komputerowi jedynie, z jakim konkretnym elementem struktury tekstu ma do czynienia i pozostawiamy mu pełną swobodę decyzji co do tego, jak ten element zostanie przez niego zaprezentowany w finalnym składzie – jak będzie wyglądał na stronie. Nie wskazujemy miejsca na stronie, gdzie tekst przypisu ma się pojawić, nie dbamy o nadanie mu odpowiedniego numeru, zastosowanie odpowiedniego stopnia pisma ani ewentualnych dodatków estetycznych. Określamy tylko kategorię tego fragmentu tekstu.

Niestety żadna róża nie jest pozbawiona kolców. Tak też jest i w tym przypadku. Owszem, przypis możemy wstawić stosunkowo łatwo, ale modyfikacja tego, jak ten przypis wygląda, może już nie być taka prosta. Dużo zależy od twórcy makra, z którego korzystamy. Mógł on przewidzieć, że użytkownikom będzie potrzebna możliwość dokonywania takich zmian, jak np. zmiana symboliki, którą oznaczane są przypisy, czy zmiana ich pozycji na stronie. Jeśli tak, to w tym przypadku najczęściej wystarczy tylko zmiana określonego parametru bądź parametrów. Jeśli nie, to użytkownik jest zdany na niewdzięczne zadanie samodzielnej modyfikacji

kodu makra. Zważywszy na to, że istota sposobu, w jaki programujemy komputery, nie zmieniła się od lat siedemdziesiątych XX wieku, często nie jest to przedsięwzięcie ani proste, ani przyjemne.

O ile autorzy **troffa** mieli łatwy dostęp do fototypu, o tyle takiego luksusu nie miał już prof. Donald Knuth. Zresztą, gdyby nawet miał dostęp do fototypu drugiej generacji, to nie byłby tym przesadnie zachwycony, ponieważ to właśnie frustracja wywołana rezultatami działania takiej maszyny stanowiła dla niego motywację do stworzenia własnego systemu do składu komputerowego. Kiedy w 1977 roku przygotowywano ponownie do druku część dzieła prof. Knutha pt. *Sztuka programowania komputera*<sup>15</sup> i przedstawiono mu szrotki, ten uznał je za ohydne (zwłaszcza w porównaniu z jakością wydania z 1968 roku złożonego za pomocą monotypu) i wkrótce napisał dla siebie notatkę, w której określił podstawowe cechy, jakie powinien mieć jego własny system do składu komputerowego. Systemowi temu nadał nazwę TeX (czytaj „tech”)<sup>16</sup>, ponieważ jako pole zastosowania tego systemu wyobrażał sobie dokumenty techniczne, szczególnie te zawierające dużą liczbę równań matematycznych. Zadanie, jakkolwiek trudne, nie wydawało się bardzo skomplikowane i prof. Knuth początkowo zakładał, że uda się je wykonać w 1978 roku. Niestety okazało

>Lorem ipsum dolor sit amet. . .<sup>1</sup>

<sup>1</sup> Consectetur adipiscing elit

Il. 2. Dokument wygenerowany ze źródła umieszczonego w tekście za pomocą programu groff (jednej z odmian roffa)

się, że liczba problemów, które należy rozwiązać, i technologii, które trzeba najpierw opracować od zera, sprawiły, że TeXa nie można było uznać za ukończony aż do 1989 roku. Rezultaty jednak, szczególnie dla środowisk technicznych, były warte oczekiwania. System TeX jest aktywnie używany i rozwijany do dziś, i do dziś nikomu nie udało się stworzyć żadnego rozwiązania, które byłoby choć w przybliżeniu tak wygodne w składzie równań matematycznych.

W TeXu zaimplementowano m.in. bardzo zaawansowany mechanizm automatycznego łamania tekstu (dużo bardziej wyrafinowany niż ten obecny w **roffie**), obsługę grafiki, koloru czy wirtualnych czcionek (do tego celu stworzono cały system, który umożliwiał tworzenie czcionek komputerowych). Podobnie jak **troff**, również TeX pozwala użytkownikowi na rozszerzanie możliwości systemu dzięki makrom. Z czasem niektóre zestawy makr rozrosły się do tego stopnia i stały się tak popularne, że można było je uznać za osobne systemy. Najpopularniejszymi dzisiaj są LaTeX i ConTeXt. Zarówno jeden, jak i drugi pozwala na łatwe wstawianie do tekstu przypisów i dość daleko idącą modyfikację ich parametrów prezentacyjnych. Wynik przetworzenia tekstu:

```
\documentclass{article}
\usepackage[utf8]{inputenc}
\usepackage{polski}

\begin{document}

\section{Przypisy w \LaTeX{u}}

Wstawianie przypisów w \LaTeX{u} jest bardzo proste\
footnote{Przypisy działają należycie.}

Treść przypisu można zapisywać bezpośrednio w tekście,
którego przypis dotyczy\footnote{Tak jak tutaj.} ale także
poza nim\footnotemark{} w osobnej linii.

\footnotetext{Ten przypis był zapisany poza tekstem, do
którego się odwoływał.}

Można też kilkukrotnie\footnotemark{} odnieść się do
tego samego przypisu\footnotemark[ $\value{footnote}$ ].
\footnotetext{Przypis z dwoma odniesieniami.}
\renewcommand{\thefootnote}{\roman{footnote}}
Również zmiana numeracji na numerację rzymską\
footnote{To jest przypis z odnośnikiem będącym liczbą
```

```
rzymską.} nie jest wcale trudna\footnote{Ani skomplikowa-
wana.}
\end{document}
```

przez program LaTeX można zobaczyć na ilustracji 3. Tutaj również widzimy, że autor tekstu jedynie wskazuje komputerowi, które fragmenty tekstu należy opatrzyć przypisem o danej treści, a komputer decyduje już o całej reszcie. Podany przykład ujawnia też przy okazji kilka innych interesujących możliwości, które system LaTeX (a właściwie TeX) posiada odnośnie do zarządzania przypisami, a których próżno szukać nawet w nowych systemach składu.

Dokument LaTeXa zaprezentowany przed chwilą doskonale uzasadnia tytuł tej części artykułu. Patrząc na jego poszczególne fragmenty – tekst wymieszany z instrukcjami dla komputera – trudno oprzeć się wrażeniu, że oglądamy źródło programu komputerowego. Jest tak w istocie ze wszystkimi tego pozytywnymi i negatywnymi konsekwencjami. Do pozytywów należy bezsprzecznie zaliczyć elastyczność takiego rozwiązania, praca z systemem komputerowym na tym poziomie narzuca bowiem stosunkowo najmniejsze ograniczenia na swojego operatora<sup>17</sup>. Następnym plusem jest potencjał daleko idącej automatyzacji składu. Dziś systemy, takie jak Arbortext czy ReportLab, nawiązujące do traktowania tekstu

## 1 Przypisy w $\LaTeX$ u

Wstawianie przypisów w  $\LaTeX$ u jest bardzo proste<sup>1</sup>

Treść przypisu można zapisywać bezpośrednio w tekście, którego przypis dotyczy<sup>2</sup> ale także poza nim<sup>3</sup> w osobnej linii.

Można też kilkukrotnie<sup>4</sup> odnieść się do tego samego przypisu<sup>4</sup>.

Również zmiana numeracji na numerację rzymską<sup>v</sup> nie jest wcale trudna<sup>vi</sup>.

---

<sup>1</sup>Przypisy działają należycie.  
<sup>2</sup>Tak jak tutaj.  
<sup>3</sup>Ten przypis był zapisany poza tekstem, do którego się odwoływał.  
<sup>4</sup>Przypis z dwoma odniesieniami.  
<sup>v</sup>To jest przypis z odnośnikiem będącym liczbą rzymską.  
<sup>vi</sup>Ani skomplikowana.

1

Il. 3. Wynik kompilacji dokumentu prezentowanego w treści artykułu za pomocą systemu LaTeX

jak programu komputerowego, są szeroko wykorzystywane wszędzie tam, gdzie istnieje potrzeba generowania przyzwoitej jakości dokumentów przeznaczonych do druku, a opartych na dużej liczbie często zmieniających się danych, jak np. katalogi czy dokumentacje techniczne. Trudno oczekiwać po takich dokumentach artystycznego polotu (niestety systemy te mają też swoje ograniczenia), jednak wszystko to jest osiągalne na żądanie i bez najmniejszego dozoru. Dobrze wdrożony system tego typu jest w stanie połączyć się z odpowiednią bazą danych (także za pośrednictwem internetu), pobrać tekst, grafiki i samodzielnie wstawić je na poszczególnych stronach generowanego dokumentu. Wszystko to niemal za „naciśnięciem jednego guzika”.

Ten sposób podejścia do zagadnienia składu stwarza również problemy. Pierwszym jest spora nieprzystępność tego typu rozwiązań dla zdecydowanej większości współczesnych projektantów. Prawdopodobnie jest to związane z koniecznością pracy z dużą liczbą abstrakcyjnych struktur wyrażonych za pomocą tekstu, a nie struktur wizualnych, dużo przyjaźniejszych dla osób uzdolnionych plastycznie. Drugą przeszkodą jest zapewne czas, który upływa między zapisaniem instrukcji dla komputera a kontrolą wyników ich działania, choć obecnie długość pętli kod – interpretacja kodu – wynik jest już w dużej mierze zminimalizowana.

Wspólnym deficytem TeXa, LaTeXa i podobnych do nich systemów, w których „książka jest programem komputerowym”, jest też zmniejszająca się w miarę pracy z tekstem jego czytelność. Każdy niuans typograficzny, który trzeba uchwycić, dodatkowy firt odstęp tu i tam, to następna instrukcja. W rezultacie tekst i komendy dla komputera mogą wymieszać się tak mocno, że tekst przestaje być czytelny w sposób płynny, a to z kolei może jeszcze bardziej utrudniać i tak już często nieproste nanoszenie korekty czy zwykłe modyfikacje. Nie znaczy to absolutnie, że przez to TeX czy ReportLab stają się zupełnie nieużyteczne. Możliwość głębokiej modyfikacji ich działania, choć niełatwa, jest w dalszym ciągu niedościgniona dla kolejnej kategorii rozwiązań, którymi się zajmujemy, a których geneza i ewolucja sprawiła, że są niemalże naturalnie odporne na bolączki systemów wysoce zautomatyzowanych.

## WYSIWYG

Jeżeli chcielibyśmy określić zbiór charakterystyk wyróżniających dobrego operatora maszyny składającej, to poza dokładnością i sumiennością musielibyśmy również wymagać od niego takiej cechy jak sprawna wyobraźnia. W dużej mierze

jest to też prawdą w przypadku systemów, takich jak **troff** czy TeX, a związane jest m.in. z tym, że między decyzjami i działaniami podjętymi przez człowieka a możliwością kontroli ich skutków musi upłynąć pewien czas. Różni się w zależności od konkretnej metody – operator monotypu musiał czekać zdecydowanie dłużej na sprawdzenie wyników swoich starań niż np. osoba pisząca artykuł w TeXu – niemniej jednak czas ten pozostaje niebagatelny. Nie jest zatem niczym zaskakującym, że ludzie w sposób naturalny dążą do skrócenia pętli między działaniem a jego rezultatem.

Czynnikiem wymagającym interwencji wyobraźni jest także pewien dysonans operacyjny między czynnością, lub raczej: tworzywem tej czynności a jej wynikiem. To nieco enigmatyczne określenie można zilustrować bardzo prosto: żeby spowodować wstawienie między akapitami tekstu pionowego odstępu o długości 12,7 mm, operator LaTeXa napisze przykładowo `\space{12.7mm}`. Tekst komendy jednak ma się nijak do odstępu, który powstanie finalnie na stronie. Operator musi **wyobrazić sobie**, ile to jest 12,7 mm i jak działa `\space`, zanim w ogóle tej komendy użyje. Do tego potrzebne jest aktywne działanie intelektu, które znacznie trudniej niż prosta koordynacja typu ręka–oko przekłada się w wykształcanie odruchów przyspieszających pracę i czyniących ją bardziej wydajną i wygodną, nie wspominając już o mniejszym potencjale popełnienia błędu.

Zarówno problem związany z czasem, jak i ze ścisłym rozdziałem materii działania i rezultatu pchały ludzi w kierunku poszukiwania rozwiązań, które wyeliminowałyby lub przynajmniej ograniczyły wspomniane trudności w interakcji między człowiekiem a maszyną. Czyż nie prościej byłoby wskazać na ekranie fragment tekstu, a potem dosłownie przesunąć go w żadaną stronę niż wpisywać komendy? Jednym z pierwszych rozwiązań do zastosowań profesjonalnych, które realizowało w pewnym stopniu to marzenie, był Satz-Gestaltungs-Terminal, w skrócie SGT. Był to opracowany w drugiej połowie lat siedemdziesiątych, a dostępny komercyjnie w latach osiemdziesiątych XX wieku komputer współpracujący z fotoskładarką CRTronic, który oferował dość dokładny podgląd przygotowywanego dokumentu jeszcze przed przeniesieniem go na światłoczułe medium. Broszura reklamowa wspomina nawet o możliwości bezpośredniej edycji strony widocznej na ekranie, choć SGT nie miał innych urządzeń wejściowych poza klawiaturą, więc o „wskazywaniu elementów na ekranie” nie mogło być mowy. To już było domeną prototypowych systemów Xeroxa, a następnie ogólnodostępnych komputerów firmy Apple korzystających m.in. z wcześniejszych koncepcji i wynalazków



opracowanych oraz wstępnie przetestowanych w założonym przez Uniwersytet Stanforda instytucie SRI<sup>18</sup>. Te właśnie maszyny spopularyzowały tzw. graficzny interfejs użytkownika, czyli pracę z komputerem opierającą się nie na strumieniu tekstowych komend i odpowiedzi na nie, ale na wizualnej manipulacji. Z kolei paradygmat, polegający na dążeniu do jak największej zgodności między tym, co człowiek obserwuje na ekranie, a tym, co uzyska np. na kartce czy fotoskładowej folii, zaczęto określać akronimem WYSIWYG, od angielskich słów *What You See Is What You Get*<sup>19</sup>.

Koniec lat osiemdziesiątych i początek lat dziewięćdziesiątych przyniósł ze sobą dalszy rozwój interfejsów graficznych i pewne okrzepnięcie wcześniej rewolucyjnych i nowatorskich idei. Następnym logicznym krokiem było przeniesienie zastosowań bardzo drogiej, wysoce wyspecjalizowanych i przez to mocno nieelastycznych komputerów związanych z konkretnymi urządzeniami fotoskładowymi na powszechnie dostępne, znacznie uniwersalniejsze i stosunkowo tańsze komputery tzw. osobiste. Była to kontynuacja tendencji (królującej do dziś), żeby jak najwięcej kompetencji przenosić do oprogramowania. W wyniku tego kierunek ewolucji maszyn do fotoskładu doprowadził je do punktu, w którym nazywanie ich w dalszym ciągu

fotoskładarkami stało się nonsensowne. Zamiast tego przybrały nazwę naświetlarek, gdyż pozbawione matryc, obrazów czy nawet cyfrowych opisów czcionek zaczęły służyć wyłącznie do naświetlania folii<sup>20</sup> lub, w późniejszym okresie, bezpośrednio płyt offsetowych<sup>21</sup> według danych przesyłanych wprost z komputera. Ta sama tendencja spowodowała w latach dziewięćdziesiątych powstanie zjawiska DTP, czyli **Desktop Publishing**, oznaczającego „publikowanie prosto z biurka” (na którym stoi używany do tego celu komputer).

Od początku ambicją aplikacji DTP, opartych w sposób niemal naturalny na paradygmacie interfejsu graficznego, była także realizacja idei WYSIWYG. Pierwsze narzędzia DTP dawały rezultaty mizerne, jeśli odnieść je do dzisiejszych standardów, ale jak na owe czasy znakomite. Łatwość i intuicyjność wizualnego modyfikowania projektu, jak każde świetne rozwiązanie, jest o tyle wspaniała i czyni pracę dużo wygodniejszą, o ile też zniewalająca. Zniewalająca, ponieważ na pewien czas daje ludziom złudne poczucie, że oto znaleźli najlepsze z możliwych rozwiązań, do którego, jak do idealnego dzieła sztuki, nic już nie można dodać ani nic też z niego ująć. Szczęśliwie każdemu upojeniu towarzyszy również otrzeźwienie, wytwarzające pewne

apetyty, które z biegiem czasu tylko rosną. W przypadku DTP bardzo szybko okazało się, że określenie, gdzie ma się znaleźć dany element, nie wystarczy – potrzeba czegoś więcej. Potrzeba pewnej dozy automatyzacji, a co za tym idzie, umieszczenia projektu na osnowie podbudowującej go struktury. Owszem, od początku uznawano za oczywiste, że tekst ma być automatycznie formowany w wiersze i lepiej lub gorzej łamany, tak aby człowiek nie musiał wskazywać i ustawiać obok siebie poszczególnych liter. Niemniej jednak nie wszystkie rozwiązania dostępne dzisiaj były już na tym wstępnym etapie oczywiste. Dość szybko wytworzyło się pojęcie **stylu**, czyli formuły grupującej parametry prezentacyjne np. tekstu, która to formuła może zostać powiązana z odpowiednim obiektem i sprawiać, że np. wszystkie fragmenty tekstu, do których odpowiedni styl przypisano, będą składane tym samym krojem, o tym samym kolorze i tym samym stopniu pisma. Jednocześnie zmiana definicji stylu pociąga za sobą automatyczną zmianę parametrów związanych z nim fragmentów tekstu. Widać tutaj głęboki ukłon

**Kierunek rozwoju przypisów jest zgodny z kierunkiem, którym już od jakiegoś czasu podąża publikowanie elektroniczne**

w stronę rozwiązań stosowanych w opisanych wcześniej systemach traktujących skład jako program komputerowy. Widać również zwrócenie się ku traktowaniu składu, a szczególnie tekstu, nie

tylko z perspektywy wizualnej, ale też strukturalnej, ponieważ każdy przypisany do tekstu styl rozpina go na ramach pewnej ontologii.

Choć wprowadzenie funkcjonalności stylu znacznie ułatwia w programach DTP wprowadzanie do tekstu przypisów, to jednak natura przypisu, w tradycyjnym, mocno wizualnym znaczeniu tego słowa, nie może być ani w pełni, ani nawet satysfakcjonująco opisana stylem w formie znanej z systemów DTP. Styl taki, w odróżnieniu od „stylu” np. w systemie TeX<sup>22</sup>, ogranicza się w zasadzie tylko do określenia wyglądu tekstu przypisu, zupełnie ignorując kwestie jego położenia względem tekstu głównego czy automatyzacji zarządzania odniesieniami do niego. Żeby zaspokoić te potrzeby, niezbędne jest coś zgoła innego – specjalna funkcjonalność osadzona w aplikacji do składu.

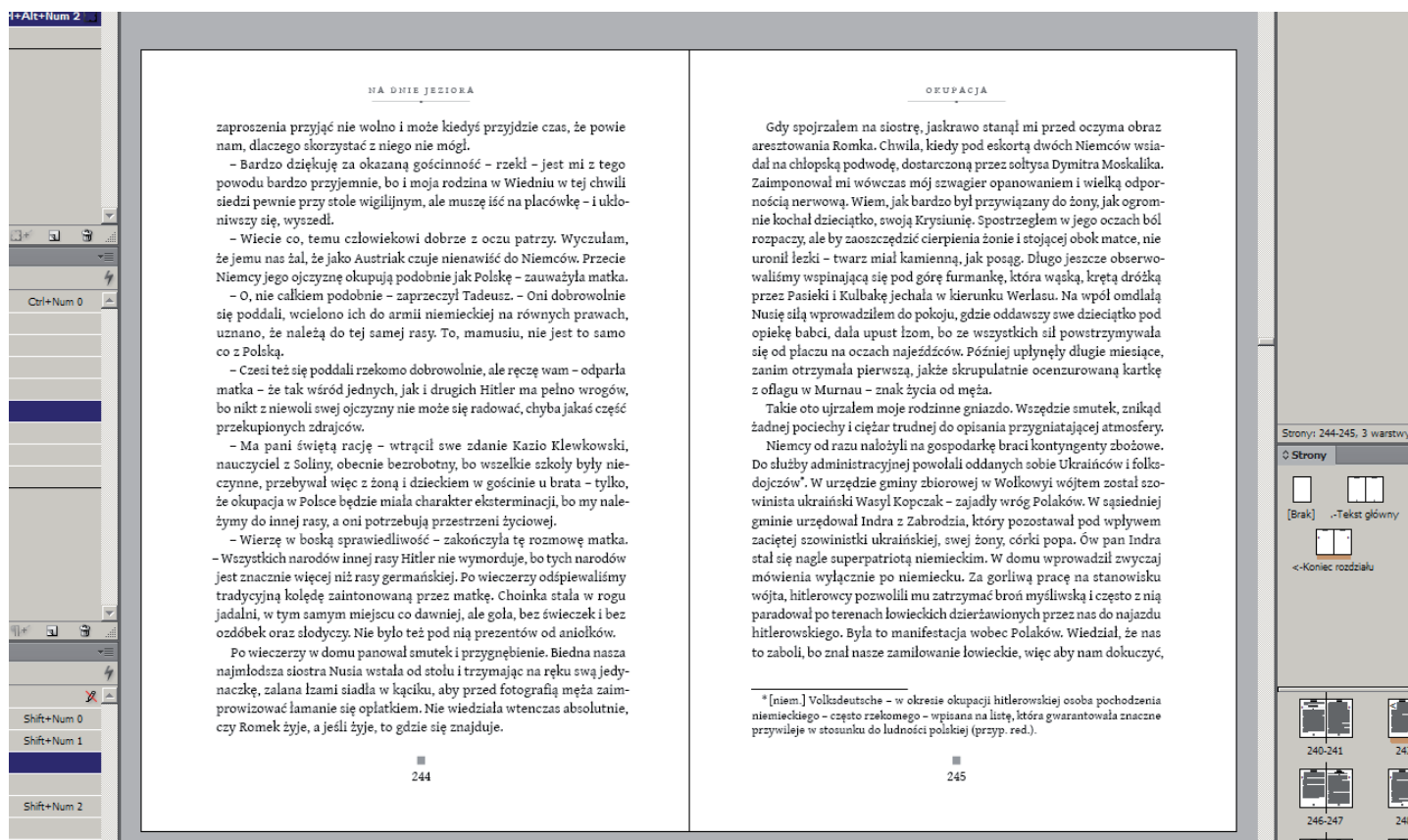
Współcześnie wśród wysokiej jakości systemów WYSIWYG do projektowania publikacji prym wiodą Adobe InDesign oraz QuarkXPress. Obydwie aplikacje dają użytkownikowi możliwość wstawienia do tekstu przypisów (il. 4), dbają o ich odpowiednią numerację (bądź oznaczenie symbolami), umieszczają u dołu kolumny<sup>23</sup> oraz pozwalają określić wygląd przypisów (il. 5), jak i linii oddzielających je

od tekstu (il. 6). Trzeba przyznać, że jest to dość spore ułatwienie w stosunku do całkowicie manualnych metod pracy. Biada jednak temu, kto zechce w swojej fantazji wykroczyć choćby odrobinę poza ramy określone scenariuszami przewidzianymi przez autorów programu. Nie musi to być wcale wykroczenie ekstrawaganckie. Wystarczy, że projektant zapra- gnie zastosować linię nieco bardziej zdobną niż standardowy, jednolitej szerokości odcinek jako separator dla obszaru zajmowanego przez przypisy dolne. W takiej sytuacji jest skazany na samodzielne umieszczenie jej w odpowiednim miejscu strony<sup>24</sup>. Podobnie problematyczne okazuje się stosowanie własnej sekwencji symboli jako odnośników do kolejnych przypisów. Wspomniane przypadki są istotnie rzadkie, ale razem nie są też niesłychane. Kiedy już projektant stanie w ich obliczu, musi uciekać się w najlepszym przypadku do trików wykorzystujących funkcje aplikacji zaprojektowane do zupełnie innych celów, a w najgorszym – do pracy ręcznej. Ta zaś jest irytująca nie tyle ze względu na swoją monotoność i potencjalną „pomyłkogenność”, ale z uwagi na podświadome przekonanie, że przecież zbudowanie właściwego, w pełni automatycznego rozwiązania nie powinno stanowić większego

problemu. Istotnie, technologia nie czyni tego niemożliwym, a każdy projektant DTP jest w stanie stosunkowo szybko i bardzo precyzyjnie określić swoje wymagania w konkretnym przypadku. Żeby jednak wprowadzić je w życie, musiałby je samodzielnie zawrzeć w formie programu komputerowego, co samo w sobie nie jest nieosiągalne, ale niestety bardzo czasochłonne i wymagające sporej wiedzy, a przez to niejednokrotnie zwyczajnie nieopłacalne.

To wszystko w połączeniu z poprzednio przedstawionymi **roffem** i TeXem pozwala nam stwierdzić, że o ile podejście WYSIWYG bije te systemy na głowę pod względem intuicyjności, bezpośredniości i łatwości użycia, o tyle w kwestii automatyzacji opartej na tworzeniu generalizacji pewnych powtarzalnych wzorców – przynajmniej współcześnie – ze wstydem im ustępuje.

Obecnie technologie DTP w tandemie z paradygmatem WYSIWYG są już poważnym standardem. Należy jednak pamiętać, że nawet przy dzisiejszym stanie technologii informacyjnych i technologii wyświetlania w akronimie WYSIWYG zawsze powinno być miejsce na dodatkowe słowo: *almost* (ang. „prawie”). W rzeczywistości bowiem *What You See Is*



Il. 4. Dokument Adobe InDesign z przypisem złożonym przez tę aplikację według zadanych parametrów

*Almost What You Get*. Łatwo o tym zapomnieć przy ekranach o szerokich gamach kolorów i coraz wyższych rozdzielczościach, ale nawet najlepszy z nich nie będzie w stanie oddać faktury papieru, uszlachetnień druku (np. różnego typu lakiery), tłoczeń, złocień itd.

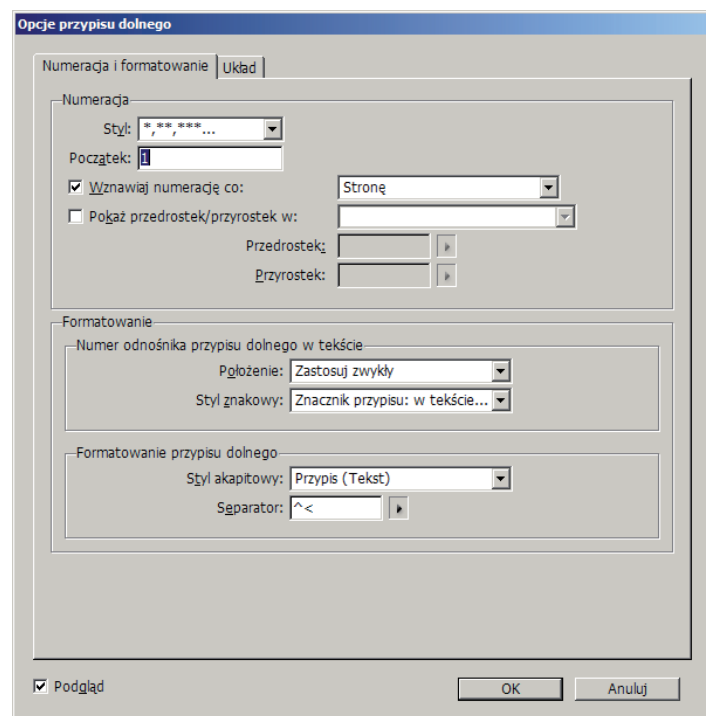
## *Libera electronicus*

Choć za pierwszą jaskółkę książki elektronicznej, a może raczej: elektromechanicznej, można uznać już koncepcję **memexu**, to o książce elektronicznej zastosowanej w praktyce i prawdziwie użytecznej można mówić dopiero z początkiem lat siedemdziesiątych ubiegłego stulecia. Był to czas powstawania systemu UNIX. Jednym z założeń projektowych tego systemu było zintegrowanie z nim jego własnej dokumentacji. W ten sposób narodził się tzw. **man** (od angielskiego *manual*), czyli **podręcznik, instrukcja**. Składał się on z programu o tej samej nazwie oraz całej biblioteki dokumentów o różnej długości opisujących działanie poszczególnych programów, komponentów systemu oraz funkcji, których można używać przy budowie oprogramowania do niego. Co prawda dokumentacja ta występowała także w wersji drukowanej, ale wraz z upowszechnianiem się terminali ekranowych, to właśnie

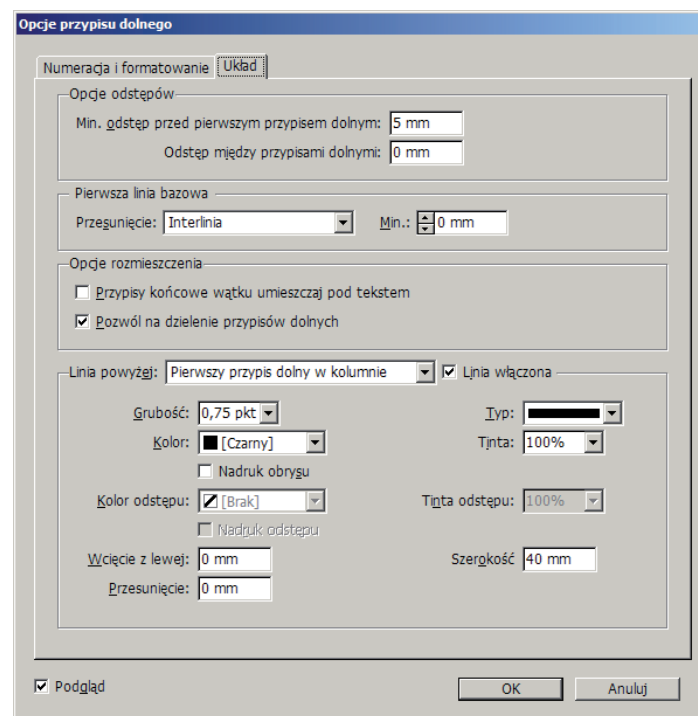
ekran stawał się głównym medium dla materiałów zawartych w **manie**. Mimo że powstanie **mana** nie wyznacza w ogólności początku praktycznej książki elektronicznej, to status kamienia milowego przysługuje mu choćby z racji tego, że jest to bodaj pierwszy tego gatunku wysiłek tak skoncentrowany i zintegrowany w sobie.

Zanim przejdziemy dalej, warto w tym miejscu zaproponować pewną definicję książki elektronicznej. Książką elektroniczną, czy też **e-bookiem**, będziemy dalej nazywać taki zbiór danych przechowywanych w formie elektronicznej (można powiedzieć: w pamięci komputerowej), który z tworzyw właściwych książce tradycyjnej jest w stanie przechowywać przynajmniej tekst, i to właśnie tekst ułożony w pewną narrację stanowi rdzeń przekazu tego zbioru. Definicja ta, mimo że w żaden sposób nie można jej uznać za kompletną czy bezwzględnie normatywną, jest na tyle pojemna, że obejmuje np. zwykły plik tekstowy, ale na tyle restrykcyjna, że wyrzuca poza swój margines większość gier komputerowych czy baz danych. Poza tym daje nam ona bazę wspomagającą naszą intuicję, mimo że nie jest to baza niewzruszona.

Następnym, bardzo znaczącym, krokiem w rozwoju książki elektronicznej jest opracowanie przez firmę Adobe najpierw formatu **PostScript**<sup>25</sup>, a potem **PDF** (*Portable Document Format* – ang. „przenośny format dokumentu”),



Il. 5. Ustawienia programu Adobe InDesign w wersji CS6 dotyczące formatowania tekstu przypisów



Il. 6. Ustawienia programu Adobe InDesign CS6 określające położenie przypisów w ramce tekstowej

korzystającego w dużej mierze z doświadczeń wyniesionych z pracy nad tym pierwszym. Główną ideą przyświecającą utworzeniu zupełnie nowego formatu, jakim był PDF, było pragnienie, żeby dokument przygotowany przy użyciu jednego systemu komputerowego, mający dzięki temu określony wygląd, po otwarciu w zupełnie innym systemie wyglądał dokładnie tak samo. Z perspektywy współczesności może się to wydawać trywialne, ale wystarczy nawet pobieżny rzut oka na stan narzędzi z początku lat dziewięćdziesiątych ubiegłego wieku, żeby zdać sobie sprawę z tego, że budowa swego rodzaju standardu, jakim później stał się PDF, nie była wcale bezproblemowa – wręcz przeciwnie<sup>26</sup>.

Dzisiaj PDF kojarzy się przede wszystkim jako *de facto* standard do przesyłania dokumentów między poszczególnymi podmiotami biorącymi udział w procesie profesjonalnego druku i druk jest obszarem, na którym PDF panuje niemal niepodzielnie. Format jednakże był projektowany do znacznie szerszych zastosowań i przez lata swego rozwoju wcale tego potencjału nie zatracił.

Chociaż PDF skupia się na precyzyjnym zachowaniu wyglądu przenoszonego dokumentu, to wcale nie znaczy, że realizacja tego celu wyznaczyła nieprzekraczalne granice rozwoju formatu. Okazało się bowiem, że jest wysoce pożądane, żeby informacje zawarte w dokumentach PDF mogły być przekształcane automatycznie. Jest wiele powodów po temu: przetwarzanie przez automatyczne systemy wyszukujące, odczytywanie tekstu przez syntezytory mowy (dla osób niewidomych) czy konwersja na inne formaty. Technikę, która jest podstawą tego potencjału, specyfikacja techniczna formatu PDF określa mianem **tagowania (znacznikowania)**. Polega ona na przypisywaniu elementom dokumentów fragmentów tekstu, niewidocznych dla użytkownika, które mogą być zorganizowane w drzewiastą strukturę. Tutaj dopiero nabierają pełnego blasku i łączą się w jedną, logiczną całość inspiracje i koncepcje opisane wcześniej: dokument nie tylko „wygląda”, ale również ma przypisaną do siebie *explicite* strukturę logiczną. Użytkownik formatu ma dużą swobodę w decydowaniu o kształcie swoistych **etykietek** tworzących tę strukturę, ale standard również definiuje podstawowy ich zestaw. Wśród nich znajduje się również **tag** oznaczający przypis<sup>27</sup>. Dzięki temu komputer może, o ile dokument zawiera **tagi** (co nie jest obligatoryjne), wykryć i potencjalnie odpowiednio obsłużyć przypisy (np. zaprezentować je w osobnym oknie).

W przypadku książki elektronicznej warto również pamiętać o tym, że urządzenia, na których książka ta będzie odczytywana, mają bardzo różne wymiary. PDF, podążając za przykładem **PostScriptu**, opiera się na pojęciu fizycznej strony – arkusza papieru – mającej konkretne wymiary i proporcje. Już we wczesnych latach PDF-a okazało się, że czasem wygodnie jest zawartość dokumentu przedstawić na innym formacie (np. dopasowanym do proporcji wyświetlacza). To znaczy, że tekst, ilustracje – słowem: wszystko, co stanowi dokument – należy „przebrać w nową formę”. Niestety nie zawsze jest oczywiste, jak dokładnie, a przede wszystkim w jakiej kolejności należy to zrobić. Niemniej jednak PDF przewiduje możliwość „przelania”<sup>28</sup> treści. Dzieje się tak również dzięki **znacznikowaniu**. Nie jest to najbardziej wyrefinowana technologia, ale i tak bije na głowę próbę czytania książki zapisanej jako PDF w formacie mającym się nijak do proporcji wyświetlacza urządzenia.

Z uwagi na ograniczone miejsce nie jesteśmy w stanie

zająć się wszystkimi aspektami publikacji przez kanały elektroniczne. Z tej samej przyczyny nie możemy opisać wszystkich, używanych dzisiaj oraz historycznych, formatów książek, wskazując ich specyfikę odnośnie do przypisu. Nie możemy sobie jednak

pozwolić na pominięcie dwóch połączonych ze sobą i chyba najistotniejszych obecnie technologii, mianowicie: technologii webowych i formatu EPUB.

Schyłek ubiegłego wieku był świadkiem narodzin internetu popularnego, dostępnego łatwo i na wielką skalę. Fundamentem internetu, przez niektórych utożsamianym przez kontekst przeglądarki z internetem jako takim, jest język HTML, wspomniany uprzednio przy okazji omawiania pojęcia hipertekstu. Język ten jest zorientowany przede wszystkim na opisanie struktury dokumentu, którą ujmuje w ramy logicznego drzewa elementów. Dopiero tak zorganizowany dokument jest interpretowany przez komputer np. w celu wizualnego przedstawienia go człowiekowi lub dalszej automatycznej transformacji. Rzut oka na standard definiujący HTML ujawnia, poza elementami mającymi sens tylko w odniesieniu do mediów elektronicznych, daleko idącą inspirację tradycyjnymi strukturami tekstu. Odnajdziemy tam elementy pozwalające na oddanie struktury nagłówków odpowiadające np. tytułom rozdziałów czy sekcji. Spotkamy elementy do formowania wycień, tabel itp. Nie znajdziemy natomiast specjalnych elementów dla przypisów. Specyfikacja mówi wprost:

Następną ścieżką rozwoju przypisów są technologie określane mianem rozszerzonej rzeczywistości

HTML nie posiada dedykowanego mechanizmu do oznaczania przypisów. Poniżej przedstawiono kilka alternatyw<sup>29</sup>.

Znamienne jest, że jako alternatywę dokumentacji wymienia znacznik `<a>`, który, notabene, służy również do zamieszczania w HTML-u tzw. **linków**, czyli **hiperodnośników** łączących treść dokumentu z wybranymi miejscami w niej samej lub osobnych dokumentach.

Podobnie jak inne przedstawione tutaj formaty i rozwiązania, również HTML ewoluował. W czasie tej ewolucji zostały z niego usunięte elementy odpowiedzialne za wygląd (np. element `<b>` nakazujący złożenie tekstu odmianą grubą), a dodane te, które lepiej przystają do logiki współczesnego internetu. Jednocześnie pojawił się mocny trend, który doprowadził do wykształcenia się dwóch uzupełniających HTML technologii: **CSS** oraz **JavaScript**. CSS (*Cascading Style Sheets* – ang. kaskadowe arkusze stylów) jest językiem,

którego zadaniem jest jednoznaczne opisanie wyglądu dokumentu HTML. **JavaScript** natomiast to język programowania o ogólnym zastosowaniu, dzięki któremu można decydować np., jak dokument ma reagować na konkretne akcje użytkownika, można nakazać komputerowi transformację danych czy wreszcie skomunikować się z innym komputerem w sieci. Wystąpił zatem trójpodział: HTML – struktura, CSS – wygląd, **JavaScript** – zachowanie. Każdy z tych języków ma inne kompetencje, wady i zalety, ale wspólnie decydują o kształcie i zachowaniu przypisów ujętych w ramy technik webowych.

Internet nie wytworzył jeszcze dla przypisów form kompozycyjnych, które można byłoby określić mianem fundamentalnie odmiennych od tych, które są uznane za tradycyjne. Najczęściej przypis w internecie zajmuje, podobnie jak w książce, miejsce u dołu „strony”<sup>30</sup>, a tym, co wyróżnia go na tle papierowego przodka, jest pewna dynamika, najczęściej sprowadzająca się do wyświetlenia niewielkiej ramki

The screenshot shows the Wikipedia article for 'Hypermedia'. A mouse cursor is hovering over a citation link, which has triggered a tooltip. The tooltip contains the following text: `http://portal.acm.org/citation.cfm?id=806036` and 'Complex information processing: a file structure for the complex, the changing and the indeterminate'. The article text below the tooltip discusses the definition of hypermedia, its history, and its application in various contexts like the World Wide Web and multimedia.

Il. 7. Strona anglojęzycznej Wikipedii dotycząca hipermediów. Po wskazaniu przypisu kursorem pojawia się nałożone na treść strony okienko z tekstem przypisu

z treścią przypisu po wskazaniu odnośnika przypisu myszą (il. 7). Często również odnośnik do przypisu jest zarazem linkiem do fragmentu strony zawierającym jego tekst (il. 8).

Podobnie sprawy mają się, gdy weźmiemy pod uwagę format EPUB zaprojektowany specjalnie z myślą o książkach elektronicznych i ich czytnikach. Jest tak w dużej mierze dlatego, że format ten jest w zasadzie zbudowany na HTML-u i CSS-ie oraz pewnych specyficznych dla siebie rozwiązaniach. Chyba największymi jego zaletami są: właściwa dla splotu HTML-a i CSS-a zdolność dopasowania się do rozmiarów urządzenia wyświetlającego, a także możliwość spięcia ze sobą wszystkich materiałów i technologii tworzących **e-booka** w pojedynczym pliku, co ułatwia jego przenoszenie i jego potencjalne uwierzytelnienie. Co prawda możliwości nadawania przypisom odpowiedniej formy zależą od możliwości konkretnego czytnika **e-booków**, ale w dużej mierze pokrywają się one z tym, co oferuje nam HTML i stowarzyszone z nim technologie w wydaniu „przeglądarkowym”. Warto przy tym oprzeć się pokusie traktowania czytników książek elektronicznych jako urządzeń zasadniczo różnych od np. tabletów

lub smartfonów. Różnica między nimi jest, poza ewentualnym użyciem wyświetlacza typu **e-ink**, bardzo powierzchowna i marketingowo-fasadowa. Marketingowa, ponieważ urządzenia te przedstawia się jako zupełnie inne rozwiązania, a fasadowa dlatego, że różnica z reguły polega tylko na zmianie pewnej frontowej aplikacji uruchamianej domyślnie wraz ze startem czytnika. Cała reszta pozostaje niemal taka sama<sup>31</sup> i dowolny smartfon lub tablet można błyskawicznie zamienić w przenośną czytelnię za pomocą odpowiedniej aplikacji.

Na zakończenie tej części przyjrzymy się jeszcze jednemu przykładowi, który na pierwszy rzut oka trudno utożsamić z przypisem, ale który ma wszelkie jego cechy. Tym przykładem są tzw. **podpowiedzi** (ang. *tooltips*) znane powszechnie z interfejsów graficznych obsługiwanych urządzeniami wskaźującymi. Co prawda nie każda aplikacja jest w nie wyposażona, ale w dużej ich liczbie możemy wskazać np. nieznaną sobie ikonę narzędzia, aby po kilku sekundach przeczytać mniej lub bardziej zwięzły opis objaśniający jego działanie czy sugerujący jego skutki (il. 9).

client interacts with the server entirely through hypermedia provided dynamically by application servers. This means that in theory no API documentation is needed, because the client needs no prior knowledge about how to interact with any particular application or server beyond a generic understanding of hypermedia. In other [service-oriented architectures](#) (SOA), clients and servers interact through a fixed [interface](#) shared through documentation or an [interface description language](#) (IDL).

---

**Cultural references** [ edit ]

*Hyperland* is a 1990 documentary film that focuses on [Douglas Adams](#) and explains adaptive hypertext and hypermedia.

---

**See also** [ edit ]

- [Cybertext](#)
- [Electronic literature](#)
- [Metamedia](#)

---

**References** [ edit ]

1. <sup>↑</sup> <http://portal.acm.org/citation.cfm?id=806036> Complex information processing: a file structure for the complex, the changing and the indeterminate
2. <sup>↑</sup> Terry K. Borsook, Nancy Higginbotham-Wheat. *A Psychology of Hypermedia: A Conceptual Framework for R&D*. 1992. Retrieved August 3, 2010.

---

**Further reading** [ edit ]

- Hicks, David L.; John J. Leggett; Peter J. Nürnberg; John L. Schnase (1998). "A hypermedia version control framework". *ACM Transactions on Information Systems*. **16** (2): 127–160. doi:10.1145/279339.279341. ISSN 1046-8188.
- Dave, Pratik; Unmil P. Karadkar; Richard Furuta; Luis Francisco-Revilla; Frank Shipman; Suvendu Dash; Zubin Dalal (2003). "Browsing intricately interconnected paths": 95. doi:10.1145/900051.900071.
- Geißler, Jörg (1995). "Surfing the movie space": 391–400. doi:10.1145/217279.215303.
- Jones, Steve (2003). *Encyclopedia of New Media*. Sage Publications. doi:10.4135/9781412950657. ISBN 9781412950657.
- Lister, Martin; et al. (2003). *New Media: A Critical Introduction* (2nd ed.). New York: Routledge. ISBN 0415223784.

---

V · T · E	Hypermedia	[hide]
<b>Basics</b>	Hypertext · Hyperlink · Hypertext fiction · Hypervideo · Adaptive hypermedia (educational · authoring) · Hyperlinks in virtual worlds	
<b>Resource identifiers</b>	Uniform resource identifier · Internationalized resource identifier · Uniform resource name · Uniform resource locator · Extensible resource identifier · Persistent uniform resource locator · Semantic URL	
<b>Concepts</b>	anchor text · click path · Domain name · click here · Copyright aspects of hyperlinking and framing · deep linking · Fat link · Fragment identifier · Hostname · Hypertext · Inline linking · inbound link/backlink · HTTP referer · image map · Internal link · Internet bookmark · linkback · Link relation · Link rot · Object hyperlinking · Path · Screen hotspot · Source tracking · transclusion · URI scheme · URL normalization · URL redirection · Website · Web page · XML namespace	
<b>Technology</b>	CURIE · Hypertext Transfer Protocol · XLink	
<b>See also</b>	Digital poetry · History of hypertext · Interactive novel · Interactive fiction · Timeline of hypertext technology · Copyright aspects of hyperlinking and framing · World Wide Web · Domain Application Protocol	

---

Categories: [Hypermedia](#) | [Hypertext](#)

---

This page was last edited on 6 April 2017, at 23:31.

Text is available under the [Creative Commons Attribution-ShareAlike License](#); additional terms may apply. By using this site, you agree to the [Terms of Use](#) and [Privacy Policy](#). Wikipedia® is a registered trademark of the [Wikimedia Foundation, Inc.](#), a non-profit organization.

[Pomóż Wikipedii](#) · [About Wikipedia](#) · [Disclaimers](#) · [Contact Wikipedia](#) · [Developers](#) · [Cookie statement](#) · [Mobile view](#)

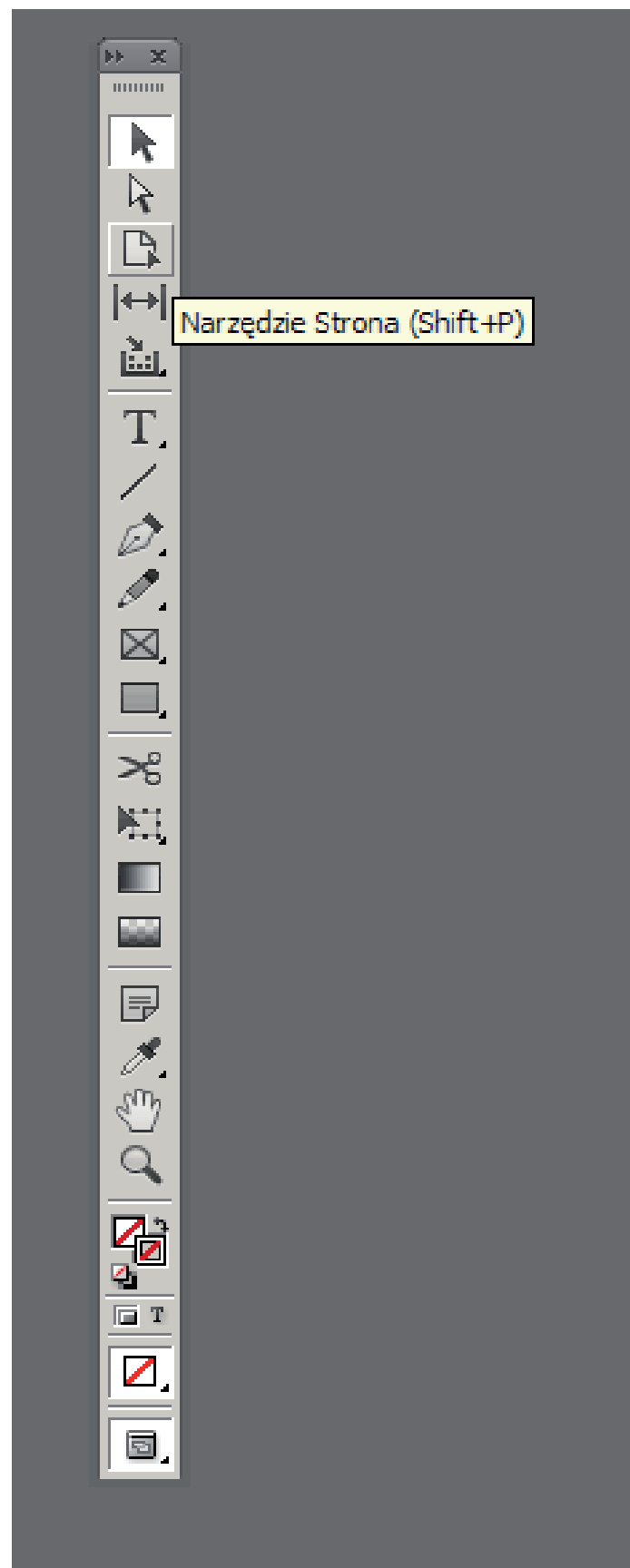
Il. 8. Strona anglojęzycznej Wikipedii o hipermediach. Po kliknięciu w odnośnik do przypisu przeglądarka kieruje użytkownika do tekstu przypisu umieszczonego u dołu strony

## Przyszłość

Tym sposobem, po przebyciu wielu poprzepłatanych ze sobą nurtów, docieramy do ostatniej, profetycznej części artykułu. Jak każde rozważania tego typu, będzie się ona opierać na projekcjach współczesności, wykonywaniu kolejnych, logicznych kroków oraz inwencji własnej autora.

Najbardziej prawdopodobny kierunek rozwoju przypisów jest zgodny z kierunkiem, którym już od jakiegoś czasu podąża publikowanie elektroniczne. Mianowicie większy udział w publikacjach mediów innych niż tekst czy statyczny obraz. Wideo i dźwięk są naturalnymi przykładami takich mediów. Tak jak mogą one służyć jako budulec integralny dla całości głównego przekazu, tak mogą też być użyte w przypisie, gdy ich związek z wiodącym wątkiem jest poboczny. Fragment filmu czy utworu może być cenniejszy niż jakikolwiek opis. Można zaprotestować, że w dzisiejszych czasach każdy zainteresowany odbiorca może samodzielnie odnaleźć odpowiednie treści np. w internecie i w ten sposób „wytworzyć” dla siebie niemal dowolny przypis. W zasadzie jest to prawda, niemniej jednak nie można zaprzeczyć wartości ewentualnego komentarza odautorskiego lub odredakcyjnego osadzającego prezentowany materiał w odpowiednim kontekście.

Wspomnieliśmy o wideo i audio jako potencjalnych nowych tworzywach dla przypisu, jednakże popełnilibyśmy zaniedbanie, gdybyśmy poprzestali tylko na nich. Poza nimi nowoczesne technologie dają nam przecież możliwość tworzenia trójwymiarowych modeli rzeczywistości, z którymi człowiek może wchodzić w wirtualne interakcje. O ilez bardziej przekonujący byłby przypis wspominający, dajmy na to, o pewnym posążku znajdującym się w zbiorach muzealnych, gdyby można było zobaczyć nie tyle zdjęcie tego posążka, ile jego model, który dałby się dowolnie obrócić, przybliżyć, a może nawet, zgodnie z wolą odbiorcy, skomponować z innymi elementami całego dzieła. Byłoby to zresztą konsygnentne z ideami **memexu**. A skoro już mowa o kontynuacji dawnych idei, interesujące rezultaty mogłaby przynieść również pełniejsza realizacja koncepcji hipertekstu, a w szczególności hipermediów, bo do tej pory jeszcze wideo czy dźwięk są zdecydowanie najczęściej traktowane jak monolityczne i niepodzielne bloki, które można co najwyżej odtworzyć, a nie połączyć z innymi mediami czy odnieść się do ich wybranych fragmentów. Można sobie bez trudu wyobrazić np. przypisy tekstowe do filmu lub nagranych dźwięku. To pierwsze w pewnym sensie już funkcjonuje jako



Il. 9. Fragment interfejsu Adobe InDesign prezentujący podpowiedź dla narzędzia (tooltip)

alternatywa dla lektora lub dubbingu w filmach obcojęzycznych. To drugie jest rzadsze, ale również występuje<sup>32</sup>.

Następną ścieżką rozwoju przypisów są też technologie określane mianem rozszerzonej rzeczywistości, to znaczy obrazu rzeczywistości uzupełnionego o dodatkowe, nałożone nań informacje. Pierwszego, bardzo prymitywnego i dalekiego praprzodka takiej technologii można by upatrywać w siatkach lotniczych celowników optycznych. Znacznie bliższe natomiast są jej i czasowo, i funkcjonalnie systemy HUD<sup>33</sup> samolotów bojowych, a ostatnio także cywilnych czy nawet samochodów. Dzięki temu, że wyświetlacz takiego systemu jest w dużym stopniu przezroczysty, pozwala on na nieprzerwaną obserwację przez pilota/kierowcę przestrzeni przed maszyną, jednocześnie nakładając na nią obraz np. sztucznego horyzontu. Odpowiednia aplikacja w smartfonie działająca na tej samej zasadzie umożliwia np. dodawanie „przypisów” do widocznych szczytów górskich czy zabytków, na które właśnie patrzy przez kamerę urządzenia człowiek. Nie jest to rozwiązanie zupełnie nowe, bo stosuje się je już od kilku lat (np. w serwisie [layar.com](http://layar.com)), niemniej jednak traktujemy je jako przyszłościowe, ponieważ trudno znaleźć jego przykłady w kontekście książki elektronicznej, a tym bardziej przypisu.

Hipotetyczne ścieżki potencjalnej przyszłości przypisu można mnożyć niemal w nieskończoność. Można je wytyczać w kierunku wirtualnej rzeczywistości czy sztucznej inteligencji. Można także eksperymentować z ich formą i dynamiką, co jest tym łatwiejsze, im bardziej dojrzałe stają się współczesne narzędzia i techniki komputerowe. Możemy np. pokusić się o stworzenie czegoś, co można byłoby określić jako **przypis blokowy**. Działałby on nie tyle na zasadzie fragmentu treści lokalizowanego za pomocą symbolu, ale jako akapit tekstu uzupełniający treść główną w sposób nieobligatoryjny. Oznacza to, że czytelnik mógłby zupełnie ignorować takie akapity, nie tylko zwyczajnie pomijając je w czasie lektury, ale w ogóle ukrywając. Równie interesujący wydaje się też eksperyment z przypisami łączonymi z tekstem nie za pomocą symbolu, ale linią wprowadzanej z miejsca odniesienia, np. na marginesie dokumentu, gdzie treść przypisu się pojawi. Przypominałoby to nieco odnośniki do komentarzy, którymi opatruje się tekst w procesorach tekstu takich jak np. Microsoft Word, ale o dużo lepszych właściwościach estetycznych.

## Podsumowanie

Chociaż przypis w elektronicznej postaci nie różni się konceptualnie od swojego tradycyjnego protoplasty, to jednak właściwe zrozumienie jego fundamentów i historii jest konieczne do zarysowania kontekstu, w jakim występuje. Bez tego nie jest możliwe określenie jego miejsca we współczesności ani wskazanie perspektyw na przyszłość. Przypis cyfrowy jest, jak obrazuje przyjęta za ramę tego artykułu rzeczna analogia, nie tyle ustalonym, statycznym obiektem, co procesem, który wypływa z określonych źródeł (nieraz niezwiązanych z nim bezpośrednio), objawia się jako efekt łączenia się ze sobą różnych prądów i inspiracji, ale jednocześnie zachowuje pewną dozę elastyczności.

**Key Words:** footnotes, history of typography, history of typesetting, DTP, typesetting, computer typesetting, TeX, troff, WWW, electronic book, e-book

**Abstract:** Contemporary footnote, and electronic footnote in particular, is a product of many diverse currents of thought and concepts – some of them being a direct continuation of typographic traditions and some without any apparent connection to them. The following article makes an attempt at exploring as great number of such concepts as possible, while keeping them in their a proper historical context and trying to emphasize the relations between them, should there be any. Starting with the methods of traditional typography and the place of footnotes in regard to these methods, we move towards early, relevant ideas underlying information revolution, onto the area of computer typography, WWW and electronic books, to finally present some potential ways in which footnotes as we know them may evolve in the future. The whole is complemented with example codes of documents in troff and LaTeX showing the specifics of working with footnotes in these systems, images of their resulting documents, screenshots illustrating how footnotes are handled by Adobe InDesign application, as well as examples of footnotes typical for WWW technologies and e-books.

<sup>1</sup> Na przykład: *American Specimen Book of Type Styles*, Jersey City 1912; *Palmer & Rey's New Specimen Book*, San Francisco 1884; *Reardon & Krebs Typographers Type Book*, San Francisco 1955; *Specimen Book and Catalogue*, Jersey City 1923.

<sup>2</sup> *Linotype Keyboard Operation*, New York 1957.

<sup>3</sup> Kuszające może się wydawać przypisanie takiej roli znakom składanym w górnym indeksie, należy jednak pamiętać, że mają one także inne zastosowania, jak chociażby skład wzorów chemicznych, równań matematycznych, ułamków czy, jak to jest np. w języku angielskim, sufiksów do liczebników porządkowych wyrażonych cyframi indo-arabskimi.

<sup>4</sup> Vannevar Bush jest postacią równie interesującą, co współcześnie mało znaną. Swoją karierę rozpoczął jako projektant i konstruktor komputerów analogowych, w której to dziedzinie odniósł spore sukcesy. Po wybuchu drugiej wojny światowej, dzięki swoim kontaktom w świecie nauki i polityki, Vannevar Bush stał się ważną postacią stojącą za utworzeniem organizacji National Defence Research Committee (NDRC), która miała koordynować wysiłki



nauki amerykańskiej w kierunku opracowywania nowych i ulepszenia istniejących rozwiązań militarnych. Przez pewien czas był przewodniczącym NDRC, a po utworzeniu Office of Scientific Research and Development (nadrzędnej wobec NDRC) objął kierownictwo tego biura. Z racji swojej funkcji Vannevar Bush był również mocno i bezpośrednio zaangażowany w projekt Manhattan, którego skutkiem było opracowanie broni nuklearnej.

<sup>5</sup> V. Bush, *As We May Think*, „The Atlantic Monthly” 1945, No. 7 (176), s. 101–108.

<sup>6</sup> Co prawda Vannevar Bush w swoim artykule mówi o przypadkowości nazwy memex, ale można podejrzewać, że wywodzi się ona od angielskiego *memory expander*, w wolnym tłumaczeniu: „poszerzacz pamięci”.

<sup>7</sup> Theodor Nelson jest twórcą tzw. Projektu Xanadu, rozpoczętego w latach sześćdziesiątych ubiegłego wieku, a inspirowanego w dużej mierze ideami Vannevara Busha. Niestety projekt nie został uwieńczony sukcesem i nie doczekał się praktycznej realizacji, ale wytworzył zestaw założeń i zasad, które miałyby spełniać przyszłościowa sieć komputerów. Współczesną sieć WWW można uznać za wytwór podobny do Projektu Xanadu, ale jednak mocno w stosunku do tego drugiego ograniczony.

<sup>8</sup> Tim Berners-Lee jest autorem pierwszych wersji protokołu HTTP, języka HTML oraz towarzyszących im technologii, które w rezultacie tworzą w pełni funkcjonalny prototyp czegoś, co dzisiaj określamy jako World Wide Web. Kariera WWW rozpoczęła się w CERN-ie pod koniec lat osiemdziesiątych i na początku dziewięćdziesiątych XX wieku. Szybko rozwijające się technologie telekomunikacyjne spowodowały błyskawiczną popularyzację tego rozwiązania, które w szybkim tempie urosło do rangi ogólnoświatowego fenomenu.

<sup>9</sup> Od *typesetter roff*, czyli „roff dla (foto)składu”.

<sup>10</sup> Wersja Kerninghana zwana jest też niekiedy *ditroff* od *device independent troff* (troff niezależny od urządzenia).

<sup>11</sup> Dokładniej: na specjalnych dyskach, które niewiele miały wspólnego z dyskami we współczesnym rozumieniu.

<sup>12</sup> Zasada działania takiego ekranu jest identyczna jak ta wykorzystywana w starym typie telewizorach czy monitorach – wiązka elektronów uderzająca w fosforującą powłokę ekranu powoduje jego lokalne świecenie.

<sup>13</sup> J. Condon, B. W. Kernighan, K. Thompson, *Experience with the Mergenthaler Linotron 202 Phototypesetter, or, How We Spent Our Summer Vacation*, Bell Laboratories Technical Memorandum, 1980, January 6, s. 1–11.

<sup>14</sup> Nieco enigmatyczne .FS i .FE oznaczają odpowiednio Footnote Start i Footnote End.

<sup>15</sup> D. E. Knuth, *The Art of Computer Programming*, Boston 1968.

<sup>16</sup> Nazwa TeX w istocie składa się z greckich liter *tau*, *epsilon* i *chi*, ale w czasie, gdy powstało auto-memorandum prof. Knutha, nie można było ich zapisać w komputerze.

<sup>17</sup> Staje się to bardziej oczywiste, jeśli porównamy ze sobą systemy takie jak TeX i systemy WYSIWYG.

<sup>18</sup> Akronim SRI oznacza centrum badawcze Stanford Research Institute. Znaczącą postacią owego centrum był Douglas Engelbart, uznawany m.in. za twórcę myszy komputerowej oraz znany jako autor prezentacji określanej często jako „matka wszystkich dem”.

<sup>19</sup> Można to przetłumaczyć jako „to, co widzisz, jest tym, co dostaniesz/uzyskasz”.

<sup>20</sup> W procesie CtF – Computer-to-Film.

<sup>21</sup> W procesie CTP – Computer-to-Plate.

<sup>22</sup> W systemach tego typu nie używa się pojęcia stylu, choć w pewnej mierze wybrane instrukcje sterujące są z punktu widzenia ich rezultatów podobne do stylów. W ogólnym przypadku są jednak dużo bardziej elastyczne i dają znacznie większe możliwości niż styl z dowolnego programu DTP. Zachowują się one bardziej jak funkcje z języków programowania niż jak „spis parametrów”.

<sup>23</sup> Program QuarkXPress obsługuje również przypisy końcowe.

<sup>24</sup> Bardziej zaawansowani użytkownicy programu InDesign mogą posilkować się funkcjonalnością tzw. obiektu zakotwiczonego, jest to jednak tylko połowicznie satysfakcjonujące rozwiązanie, bo w dalszym ciągu wymagające interwencji przy jakiegokolwiek zmianie liczby przypisów na danej stronie.

<sup>25</sup> Zasadniczo PostScript jest nie tyle formatem danych, co językiem programowania. Faktem jednak jest, że konkretne zastosowanie, do którego był od początku przeznaczony, sprawia, że określenie „format” nie jest nieuprawnione.

<sup>26</sup> Z pewnością czytelnicy pracujący w tamtych czasach z komputerem i dostępnymi ówczesnie procesorami tekstu pamiętają częste rozczarowania związane z tym, że pieczołowicie przygotowany dokument po przeniesieniu do innego komputera w celu jego wydrukowania „rozpadał się” na oczach zdruzgotanego autora na czynniki pierwsze. Niestety również i dzisiaj nie jest to sytuacja niesłychana, szczęśliwie jednak stosunkowo rzadka.

<sup>27</sup> *PDF Reference, Version 1.7*, 2006, s. 850, 855 i 903.

<sup>28</sup> W dokumentacji określone jest to jako reflow.

<sup>29</sup> Oryginalny tekst umieszczony w specyfikacji HTML5 (<https://www.w3.org/TR/html5/>) w sekcji 4.12.5 brzmi następująco: „HTML does not have a dedicated mechanism for marking up footnotes. Here are the suggested alternatives”.

<sup>30</sup> Interesujące jest to, że upowszechnienie się komputerów i witryn webowych spowodowało na masową skalę powrót książki do formy zwoju. Czymże bowiem jest strona internetowa ujmująca swoją treść w ramy określonej szerokości, ale w pionie stanowiąca przeciez

krótszą lub dłuższą „taśmę”, której tylko wybrana część jest w danym momencie widoczna na ekranie?

<sup>31</sup> Dobrym przykładem ilustrującym ten fakt jest czytnik Sony PRS-T1, który dzięki swojej sporej popularności doczekał się oprogramowania opartego na firmowym, ale niezależnego od producenta. Oprogramowaniem tym okazał się dominujący obecnie system Android z drobnymi zmianami i odpowiednio przystosowaną do zadań czytnika aplikacją działającą non stop jako interfejs graficzny całości. Niewielkie modyfikacje pozwoliły na używanie czytnika jako standardowego tabletu po uwzględnieniu specyfiki ekranu e-ink.

<sup>32</sup> Przykładem tego jest chociażby system komentarzy w serwisie soundcloud.com.

<sup>33</sup> Nazwa jest akronimem angielskich słów *Head-Up Display*, co można przetłumaczyć: „wyświetlacz na poziomie oczu”.