Narzędzia

BibTEX – narzędzie do przygotowania bibliografii

Andrzej M. Borzyszkowski

1. Czym jest, a czym nie jest BibTEX

BibTEX jest przeznaczony w zasadzie dla użytkowników systemu TEX, którzy

- przygotowują różne rozprawy, choćby naukowe, w których występuje zjawisko cytowania; na ogół są to książki czy artykuły, ale niekoniecznie, spójrz na pozycje [btxa, btxb];
- używają LATEX-a (istnieją jednak zestawy makr plainTEX-a pozwalające skorzystać z możliwości BibTEX-a).

Możliwości BibTEX-a kryją się w obsłudze baz, głównie bibliograficznych, przechowywanych w postaci *plików tekstowych*. BibTEX jest w stanie przygotować plik z bibliografią gotowy do użycia w LATEX-u.

BibTEX jest częścią systemu TEX. Nie jest on jednak makrem, stylem, formatem ani niczym innym TEX-owym – jest normalnym programem. Autorem BibTEX-a jest Oren Patashnik, sam program jest omówiony w książce Lamporta [Lam86], dokładniej w [Pat88a], a o [Pat88b] będzie jeszcze mowa. Ostatnie dwie pozycje są zawarte w każdej dystrybucji TEX-a, [btxb].

Źródłem informacji o BibTEX-u może być też artykuł Piotra Bolka w Wirtualnej Akademii, [Bol96].

2. Jak działa BibTEX

Zasada działania BibTEX-a opisana jest w książce Lamporta [Lam86, 4.3.2, str. 74]. My zaczniemy od przypomnienia zasad działania LATEX-a. Załóżmy, że piszemy pewien artykuł, którego główny plik nazywa się `manual.tex`. W artykule tym kilkakrotnie cytujemy różne pozycje, np. dwukrotnie cytujemy oryginalną dokumentację BibTEX-a autorstwa Patashnika. W tym celu gdzieś w końcu źródłowej wersji artykułu powinien się znaleźć tekst podobny do poniższego:

```
\begin{thebibliography}{Xxx99x}

\bibitem[Pat88a]{btxdoc}
Oren Patashnik.
\BibTeX{}ing. Documentation for general
   \BibTeX{} users, 8~II 1988.

\end{thebibliography}
```

gdzie `btxdoc` jest naszą nazwą dla cytowanej pozycji. W tekście źródłowym, za każdym razem gdy chcemy odwołać się do tej pozycji, używamy polecenia `\cite{btxdoc}`. Teraz LATEX działa tak:

- odczytuje tekst z pliku `manual.tex`;
- odczytuje pomocnicze dane z różnych plików; na razie interesować nas będzie plik `manual.aux`;
- składa tekst produkując plik `manual.dvi`;
- zapisuje różne informacje do dodatkowych plików; tutaj interesować nas będzie wyłącznie plik `manual.aux`.

Utworzony plik `manual.aux` zawierać będzie m. in. wiersze

```
\citation{btxdoc}
\citation{btxdoc}
\bibcite{btxdoc}{Pat88a}
```

Pierwsze dwa są skutkiem dwukrotnego użycia `\cite{btxdoc}` w pliku `manual.tex`, trzeci powstał w momencie przeczytania danych bibliograficznych i ustalenia, że etykietą dla pozycji nazwanej przez nas `btxdoc` jest `Pat88a`. W pierwszym przebiegu LATEX-a nie ma jeszcze pliku `manual.aux`, brakuje więc ostatniej informacji. W kolejnych przebiegach, w miejscach odwołań do wspomnianej pozycji, użyta będzie ta etykieta.

Ręczne przygotowanie pliku z bibliografią jest nieco męczące. Trudno jest utrzymać jednolity styl cytowania różnych pozycji, trudno jest też dopilnować, by w bibliografii *nie* znalazły się pozycje, które nie są cytowane.

BibTEX służy do automatyzacji procesu przygotowania pliku bibliograficznego. Korzysta on z pliku `manual.aux` i generuje plik `manual.bbl` zawierający bibliografię. Oczywiście musi skorzystać z zewnętrznego źródła informacji o cytowanych pozycjach.

Aby móc skorzystać z BibTEX-a należy w miejscu, gdzie powinna znaleźć się bibliografia, *dokładnie jeden raz* użyć sekwencji podobnej do poniższej:

```
\bibliography{manual,btxdoc}
\bibliographystyle{plalpha}
```

Pierwsze polecenie mówi, z jakich baz bibliograficznych należy skorzystać – a może być ich kilka

– aby znaleźć cytowane pozycje. W powyższym przykładzie będą to pliki `manual.bib` i `btxdoc.bib`.

Drugie polecenie mówi, jaki styl bibliograficzny należy zastosować przygotowując bibliografię. Styl steruje:

- uporządkowaniem pozycji bibliograficznych;
- typem etykiet pozycji (liczbowe lub tekstowe);
- pisownią imion autorów (pełna lub skrócona);
- itd.

Wówczas, po zakończeniu działania LaTeX-a, w pliku `manual.aux` znajdą się dodatkowo wiersze

```
\bibdata{manual,btxdoc}
\bibstyle{plalpha}
```

Następnie wywołujemy BibTeX-a

```
  bibtex manual
```

Nie wolno tu podać rozszerzenia aux, jest ono domyślne, por. dodatek B.2.

Podczas działania BibTeX czyta pliki

- manual.aux,
- manual.bib i btxdoc.bib, oraz
- plalpha.bst.

Wynikiem będą dwa nowe pliki: plik zawierający bibliografię, `manual.bbl`, oraz sprawozdanie z wykonanych prac, czyli tzw. log, `manual.blg`.

Plik z bibliografią będzie zawierać fragment podobny do przygotowanego ręcznie. W naszym przykładzie będzie to:

```
\begin{thebibliography}{Pat88b}

\bibitem[Pat88a]{btxdoc}
Oren Patashnik.
\newblock {{\BibTeX ing}}.
\newblock Documentation for general
  {\BibTeX} users, 8~Luty 1988.

\end{thebibliography}
```

Nic nie stoi na przeszkodzie, aby modyfikować tak przygotowany plik. Lamport nawet to tego zachęca: „while BibTeX gets … right, it is only a computer program, so …” (kilka słów na ten temat później).

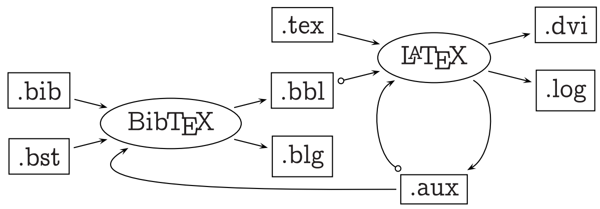
Sprawozdanie zawiera kilka informacji o wykonaniu programu oraz o ewentualnych błędach:

```
This is BibTeX, Version 0.99c (Web2C 7.3)
The top-level auxiliary file: manual.aux
The style file: plalpha.bst
Database file #1: manual.bib
Database file #2: btxdoc.bib
```

Teraz dopiero zaczyna się odliczanie podwójnego przebiegu LaTeX-a. Drugi przebieg będzie pierwszym, w którym LaTeX zauważy istnienie pliku `manual.bbl` i potraktuje go tak samo, jak gdybyśmy napisali `\input manual.bbl` w miejscu polecenia `\bibliography`. Bibliografia staje się więc rozdziałem o tytule „References” (lub innym, jeśli skorzystamy z możliwości `\def\refname{Literatura cytowana i nie}`), a `manual.aux` zawiera już informację

```
\bibcite{btxdoc}{Pat88a}
```

Biorąc pod uwagę, co też BibTeX potrafił stworzyć w pliku `manual.bbl`, żałować należy, że nie uzupełnił od razu pliku `manual.aux` – oszczędziłoby to jednego przebiegu LaTeX-a. Objaśnienia te zilustrować można poniższym rysunkiem.



Nazwy plików mają standardowe końcówki. Oprócz nazw znanych każdemu użytkownikowi TeX-a, występują też pliki o końcówkach:

`aux` plik tworzony i używany przez LaTeX-a, zawierający m.in. dane pozwalające LaTeX-owi ustalić odpowiedniość pomiędzy etykietami użytkownika a ostatecznymi etykietami pozycji bibliograficznych; jest też czytany przez BibTeX-a w celu ustalenia cytowanych pozycji bibliograficznych;

`bbl` plik z bibliografią dla danego artykułu; jest czytany przez LaTeX-a; jest przygotowywany przez BibTeX-a, choć można go też przygotować ręcznie;

`bib` plik zawierający bazę bibliograficzną;

`bst` plik zawierający styl BibTeX-owy;

`blg` log BibTeX-owy.

3. Jak wywołać BibTeX-a

Sposób uruchamiania BibTeX-a może zależeć od platformy i instalacji. Można go wywoływać z linijki rozkazowej, być może za pomocą aliasu czy pliku wsadowego, można mieć ikonę na pulpicie, choć raczej nie dla samego programu, a dla plików

*.aux, można wreszcie posiadać kompletne środowisko około-TEX-owe, np. Emacs z AucTEX-em, obejmujące BibTEX-a.

BibTEX wymaga dostępu do pliku zawierającego deklarowany styl. Style te mogą znajdować się w bieżącym katalogu, być wymienione wraz z pełną ścieżką, znajdować się w katalogu domyślnym lub wskazanym przez odpowiednią zmienną systemową. W podręczniku Lamporta [Lam86, 4.3.2, str. 74] opisano style występujące w każdej dystrybucji LATEX-a:

plain, unsrt, alpha, abbrv.

Jeśli planujemy wykorzystanie innych stylów, musimy zadbać o to, by BibTEX je zauważył.

Na płycie TEXLive4, w jednym tylko katalogu /texmf/bibtex/bst/beebe znajduje się 60 stylów, i nie są to wszystkie style na tej płycie. Niestety, na ogół nasza wiedza o tych stylach ogranicza się do ich nazwy, a jedyną metodą ich poznania jest eksperymentowanie. Alternatywnym rozwiązaniem jest własna twórczość, patrz rozdz. 8.

Do skomponowania jednej bibliografii można użyć kilku baz bibliograficznych. Bazy te mogą się znajdować w bieżącym katalogu (gdy są stworzone *ad-hoc*, na użytek jednej pracy), być wymienione wraz z pełną ścieżką, albo znajdować się w jednym z katalogów domyślnych lub wskazanych przez odpowiednią zmienną systemową. Kolejność wymienienia nazw baz gra pewną rolę, jednak tylko w sytuacjach błędogennych, por. dodatek B.2, albo w zaawansowanych zastosowaniach, por. rozdz. 7.2. Wspomniana przykładowa baza danych zawarta w każdej dystrybucji TEX-a jest oczywiście tylko przykładowa – każdy autor używający BibTEX-a będzie miał swoje własne bazy i należy umożliwić BibTEX-owi dostęp do nich.

BibTEX dopuszcza tylko jedną bibliografię w jednym dokumencie, mimo że LATEX dopuszcza kilkakrotne użycie środowiska thebibliography. Jeśli np. chcemy mieć osobno książki i osobno artykuły, publikacje i rękopisy, albo osobną bibliografię w każdym rozdziale, wówczas trzeba to zrobić ręcznie – albo wywołując kilkakrotnie BibTEX-a z różnymi bazami, albo LATEX-ując i BibTEX-ując fragmenty pliku odrębnie (w tym drugim przypadku pomocny może być pakiet bibunits). W obu przypadkach trzeba zmienić nazwę dla otrzymanych plików .bbl.

Jeśli chcemy, by w bibliografii znalazła się pozycja która *nie* jest cytowana, wystarczy w dowolnym miejscu tekstu źródłowego (ale już po \begin{document}) użyć instrukcji \nocite, tak jak to zrobiono z jedną z pozycji z tego artykułu (którą?).

4. Struktura bazy bibliograficznej

Struktura bazy bibliograficznej została opisana we wspomnianym już wielokrotnie podręczniku Lamporta [Lam86, Appendix B, str. 140–147]. Pomocą może też być przykładowa baza bibliograficzna /texmf/bibtex/xampl.bib. Dalsze wiadomości można znaleźć w [Pat88a]. Użytkownicy Emacsa mogą wywołać bibtex-mode i mieć pod ręką kilka przykładowych wzorców rekordów wraz z opisem znaczenia poszczególnych pól. Internauci skorzystają z wyczerpującego spisu i opisu typów rekordów i pól zawartych w [Bol96].

Baza bibliograficzna składa się z rekordów różnych typów. Każdy rekord składa się z nazwy typu poprzedzonej znakiem @ oraz z klucza i z pól o różnych nazwach i zawartościach stanowiących jego ciało. Przykładowy rekord,

```
@UNPUBLISHED{btxdoc,
   author = "Oren Patashnik",
   title = "{{\BibTeX ing}}",
   note = "Documentation for general
           {\BibTeX} users",
   month = "8~" # feb,
   year = 1988 }
```

jest typu Unpublished, jego kluczem jest btxdoc, posiada pola o nazwach author, title, note, month oraz year. Zawartością trzech pierwszych pól są napisy objęte cudzysłowami, zawartością pola year jest napis 1988, zawartością pola month jest konkatenacja stałej 8~ oraz zmiennej feb (o zmiennych czytaj w rozdz. 7.1).

- Zamiast nawiasów klamrowych, obejmujących ciało rekordu, można użyć zwykłych, byle do pary. Napis, czyli najczęściej cała zawartość pola, może być objęty parą podwójnych cudzysłowów, jak w powyższym przykładzie, ale szczerze tego odradzam, por. dodatek B.1. Lepiej użyć pary nawiasów klamrowych. Liczba, tzn. napis składający się wyłącznie z cyfr, nie potrzebuje otaczających nawiasów, nie są one jednak szkodliwe.
- Przecinek za ostatnim polem w ciągu pól w rekordzie nie jest konieczny, ale też nie jest szkodliwy.

- Typ rekordu i nazwa pola mogą być podane małymi lub wielkimi literami. Również w kluczu nie gra to roli. LaTeX jednak jest czuły na wielkość liter w kluczu, por. dodatek B.1.
- Klucz może się składać z liter, cyfr, myślnika czy dwukropka, a może i czegoś innego. Na pewno nie może zawierać przecinka, nawiasów i kilku innych znaków. Nie tylko spacje (jak w TEX-u), ale i zmiany wiersza w bazie danych nie grają żadnej roli. Nawet te pomiędzy znakiem `@` a nazwą typu rekordu.

Zawartość pola jest zwykłym tekstem, który jest niemal dosłownie kopiowany przez BibTEX-a, kopiowane są również zagnieżdżone nawiasy klamrowe i makra TEX-owe. Wyjątkiem są pola `title` oraz `author` i `editor`, por. rozdz. 5. Są też wprowadzane pewne kosmetyczne poprawki, np. napis 12-34 zostanie zamieniony na 12--34.

Kolejność pól w rekordzie, o ile ich nazwy są różne, nie gra żadnej roli. Kolejność rekordów jest znacząca tylko w zaawansowanych zastosowaniach, por. rozdz. 7.2.

Tekst w bazie bibliograficznej nie pasujący do wzorca

`@`$\langle$*typ*$\rangle\langle$*nawias*$\rangle\langle$*klucz*$\rangle\langle$*pola*$\rangle\langle$*nawias zamykający*$\rangle$

nie należy do żadnego rekordu. Daje to możliwość wprowadzania komentarzy do bazy danych, szerzej o tym w dodatku C.

Formalnie rzecz biorąc, rekord może mieć pusty klucz lub pusty ciąg pól, może też mieć powtarzające się pola o tej samej nazwie. Do rekordu bez klucza nie można się jednak odwołać, a powtarzające się pola będą zignorowane, skutkując jedynie ostrzeżeniem.

Sposób wykorzystania informacji zawartych w bazie bibliograficznej zależy od użytego stylu BibTEX-owego. Styl jest, *de facto*, programem, którego BibTEX jest interpreterem, a którego danymi są pliki `*.aux` i `*.bib`. Jest chyba jasne, iż w jednej bibliografii można stosować tylko jeden styl.

W podręczniku [Lam86, Appendix B, str. 140–147], a również w [Bol96], opisano wszystkie typy rekordów używanych przez cztery podstawowe style. Inne style mogą definiować swoje własne typy rekordów, choć nie jest to częste. Każdy styl zawiera osobną procedurę dla każdego typu rekordów w bazie bibliograficznej. I tak, tytuł książki (`Book`) będzie na ogół wypisany kursywą, ale ten sam tytuł będzie złożony całkiem zwyczajnie, jeśli pozycja zostanie zdegradowana do typu `InCollection`. Wydaje się, że wszystkie nieznane typy rekordów traktowane są tak jak typ `Misc`, ale z ostrzeżeniem.

Każdy ze stylów określa listy pól obowiązkowych i pól opcjonalnych dla każdego z typów rekordów. Pola spoza tych list są ignorowane. Powyższe cztery standardowe style nie różnią się w definicjach tych list. Jednak inne style mogą odwoływać się do pól ignorowanych przez style standardowe.

O dwóch specjalnych polach wspomnimy dalej, w rozdz. 7.1 i 7.2.

W rozdziale 7 opisano bardziej zaawansowane możliwości BibTEX-a, wymagające bardziej skomplikowanej składni.

5. Jak wpisywać dane do bazy bibliograficznej

Najpierw rekord. Najtrudniejszy jest sam początek. Trzeba podjąć dwie decyzje:

1. jaki typ rekordu wybrać, oraz
2. jaki klucz mu nadać.

Nie ma jednoznacznych odpowiedzi na te pytania. Czasami równie dobry jest typ `Unpublished` jak i `Misc`. Z drugiej strony nie ma to aż takiego znaczenia, jeśli pasują dwa typy, prawdopodobnie wszystkie style potraktują takie rekordy podobnie.

Najłatwiej jest rozpocząć pisanie rekordu używając narzędzi, np. wspomnianego już Emacsa z pakietem AucTEX. Można też stosować metodę „cut & paste", kopiując istniejące rekordy. Lepiej, jeśli z przygotowanej szczątkowej bazy zawierającej wszystkie możliwe pola dla danego typu rekordu.

Jeśli chodzi o nadawanie rekordom jednoznacznacznych kluczy, warto wypracować jednolity styl. Prostą możliwością jest imitowanie stylu alpha.bst BibTEX-a. A więc,

- jeśli pozycja ma jednego autora, to klucz złożony jest z trzech pierwszych liter jego nazwiska, plus dwie cyfry oznaczające rok publikacji plus kolejne litery alfabetu, o ile jest więcej takich rekordów;
- jeśli autorów jest więcej, to zamiast trzech pierwszych liter nazwiska do klucza wchodzą pierwsze litery wszystkich nazwisk, alternatywnie, tylko pierwszych trzech nazwisk.

Powyższe rozwiązanie ma dwie wady. Po pierwsze, takich etykiet jest zbyt mało na naprawdę dużą bazę bibliograficzną. Co prawda trzy litery i dwie cyfry dają ponad milion kombinacji, jednak większość cytownych prac pochodzi z ostatnich lat, autorów często jest dokładnie dwóch, a ich nazwiska częściej zaczynają się na literę K czy S, niż Y czy X. W końcu wygenerujemy klucze od `ks99a` do `ks99z`, gdzie K i S będą inicjałami różnych autorów, wprowadzając tylko zamieszanie.

Po drugie, rok publikacji nie jest, wbrew pozorom, jednoznaczny. Mogliśmy otrzymać wstępną wersję pracy w roku 1997, następnie raport techniczny w roku 1998, wersję konferencyjną w roku 1999, ukaże się ona w sprawozdaniach w roku 2000, a wersja czasopiśmienna będzie nosić datę 2001, jak dobrze pójdzie. Zmiany czynione w kluczu rekordu prowadzą w prostej linii do błędu, por. B.2. Brak zmian, tzn. utrzymywanie etykiety `ety97`, też raczej przeszkadza.

W profesjonalnych bazach danych stosowane są konwencje używania w kluczu pełnych nazwisk łącznie z jakimś skrótem tytułu.

Klucze muszą być jednoznaczne nawet w relacji równoważności *małe* = *wielkie*, w LaTeX-u muszą być jednak cytowane dosłownie.

Autor. Jest to najbardziej skomplikowane pole. Nazwiska autorów oddzielamy słowem and, nawet jeśli brzmi to niegramatycznie. Istnieją co najmniej dwa style pisania nazwisk, w zasadzie równoważne:

- imiona poprzedzają nazwisko,
- imiona występują po nazwisku, wówczas oddzielone są przecinkiem.

Komplikacje powstają gdy nazwisko zawiera dwa człony (jak u Duńczyków), inne elementy (von Junior Esq.), kombinacje liter i przecinków (na przykład, gdy nazwiskiem jest tak naprawdę nazwa firmy) i inne. Wówczas trzeba bardziej uważać, gęsto nawiasować odpowiednie człony nazwiska, por. [Lam86, Appendix B.1.2] i być może eksperymentować. Generalnie można powiedzieć, że format: *nazwisko, przecinek, imię* jest bezpieczniejszy i zwiększa szansę, że BibTEX prawidłowo rozpozna części składowe nazwiska.

Nazwisko jest ważne nie tylko samo przez się – służy też do stworzenia widocznej etykiety (w niektórych stylach) i ustawienia pozycji bibliograficznych w odpowiedniej kolejności. Jeśli brakuje nazwiska, np. pozycja jest numerem czasopisma cytowanego tylko dlatego, iż inne artykuły są w nim zawarte, wówczas dla osiągnięcia powyższych celów można użyć zastępczo pola o nazwie `key`. Przykładowo, w rekordzie [Exa] występuje `key={Example}`. Wynik widać okiem nieuzbrojonym.

Tytuł. Należy pisać jak po niemiecku (prawie wszystko wielką literą). Potem BibTEX użyje wielkich liter lub małych, zgodnie z użytym stylem. Nigdy nie zamieni małej litery na wielką, ale chętnie na odwrót. Jeśli chcemy, by BibTEX nie mieszał się do naszych liter, należy je umieścić w nawiasach klamrowych. A na pewno nie chcemy ingerencji BibTEX-a, gdy w tytule występuje czyjeś nazwisko. Należy wówczas ująć w nawias co najmniej pierwszą literę nazwiska, a nie zaszkodzi ująć je całe w nawias.

Inne pola. Jeśli nie potrafimy wypełnić sensownie pól obowiązkowych, należy się zastanowić, czy słusznie wybraliśmy typ rekordu. W każdym razie pola nie zadeklarowane będą traktowane jako puste napisy, a BibTEX ostrzeże nas, gdy dojdzie do użycia rekordu.

Można swobodnie dodawać własne pola, które na pewno zostaną zignorowane przez standardowe style BibTEX-a, np. `abstract`, `comment` i wedle fantazji. Wiele powszechnie dostępnych stylów BibTEX-owych wprowadza takie dotatkowe pola do dobrze znanych typów rekordów.

6. Sprawy polskie

O ile w TEX-u czy LaTeX-u już od pewnego czasu możemy swobodnie posługiwać się językiem polskim, o tyle BibTEX dopiero całkiem niedawno stał się bardziej otwarty na problemy językowe. Język polski ujawnia się w następujących aspektach:

- kodowanie liter,
- zamiana wielkich liter na małe,
- sortowanie liter,
- polskie skróty i słowa kluczowe,
- inne polskie obyczaje.

Oryginalne implementacje BibTEX-a w ogóle nie przyjmowały do wiadomości istnienia kodowania ośmiobitowego. W tym układzie były dwie tylko metody użycie polskich liter: albo notacja TEX-owa, np. ą=`\k{a}`, albo ciachowa.

Dopiero współczesne implementacje BibTEX-a potrafią sobie radzić z notacją TEX-ową. Zamienią `\L` na `\l`, trzyliterowym skrótem od nazwiska `{\L}o{\'s}` będzie „Łoś”, a odpowiadająca pozycja będzie umieszczona niedaleko po pozycji [Lam86].

Oczywiście nie ma i nie może być uniwersalnych zasad sortowania – nie jest wcale oczywiste, że \aa jest *ostatnią* literą alfabetu języka duńskiego. BibTEX uważa za jedną literę zestaw {\⟨*znaki*⟩}. W ramach tej „litery” może dojść do zamiany wielkości liter. Jeśli jest to niepożądane, należy użyć jeszcze jednej pary nawiasów. Tak właśnie zrobiono w tytule pozycji [Pat88a], por. rozdz. 4, co być może zdziwiło już uważnego czytelnika.

Notacja ciachowa nie jest znana BibTEX-owi – użycie w bazie bibliograficznej pisowni nazwiska /Lo/s da w wyniku etykietę [/Lo].

Obecne implementacje BibTEX-a są już raczej ośmiobitowe (ale np. BibTEX Mattesa, patrz dodatek A, chyba nie), w szczególności bez problemu zostanie skopiowany tekst w kodowaniu ośmiobitowym. Wraca jednak problem zamiany wielkości liter i nadal pozostaje problem sortowania. Stosuje się tutaj łata Piotra Bolka do jednej z implementacji (płyta TEXLive4: /support/polish/gustprog/bibtexpl.zip). Alternatywnym rozwiązaniem jest BibTEX Kempsona i Aguilar-Sierra (dodatek A) korzystający z zewnętrznego pliku rozwiązującego oba te problemy.

Jeśli chodzi o polszczyznę w tekście bibliografii, to zależy ona od użytego stylu, a nie od implementacji BibTEX-a. Gotowym rozwiązaniem są style Bogusława Lichońskiego (katalog /texmf/bibtex/bst/plbib/ na wspomnianej płycie), są to spolszczone wersje czterech podstawowych stylów BibTEX-a. Style te mają jednak dwie wady. Wadą wrodzoną jest to, że są to cztery style będące wersjami stylów podstawowych. Każdy inny styl, który wpadnie nam w ręce i którego zechcemy użyć, nie będzie jednym z tych czterech stylów i nie będzie miał polskiego odpowiednika. Wadą nabytą tych stylów jest zastosowana w nich polszczyzna. Można mieć wątpliwości, czy zwrot „8 Luty” jest właściwy, może raczej „8 Lutego”, a może „8 lutego”? Nie da się jednak tej sprawy załatwić przez proste podstawienie jednego słowa polskiego w miejsce jednego angielskiego. Język angielski jest pozycyjny, język polski jest fleksyjny: „8 lutego” brzmi świetnie, ale po usunięciu liczby już nie. Implementowanie zaś polskiej deklinacji i koniugacji w języku BibTEX-a nie jest zadaniem łatwym.

Problemem jest też decyzja, jakiego użyć kodowania polskich liter w tekście generowanym przez sam styl. Wspomniane polskie style używają ciachów. W zasadzie należałoby posiadać odrębne kopie stylu dla każdego możliwego kodowania polskich liter – niedopuszczalne jest przecież mieszanie różnych kodowań w tekście źródłowym. *Summa summarum*, wieża Babel w pełnej krasie.

7. Bardziej zaawansowane sztuczki BibTEX-a

7.1. Skróty czyli makra BibTEX-owe

BibTEX ma swój własny, prymitywny język makr, zwanych *abbreviation*. Jego prymitywizm polega na braku argumentów dla makr, na braku rozwinięć warunkowych i w ogóle na ich ograniczonym stosowaniu. Tak naprawdę są to zmienne typu napis. Miłą niespodzianką (po latach używania TEX-a) jest możliwość użycia w nazwie skrótu cyfry czy myślnika – wystarczy, jeśli nazwa zaczyna się od litery. Należy też pamiętać o zasadzie BibTEX-a *małe=wielkie* w odniesieniu do liter. Obowiązuje ona również dla nazw skrótów BibTEX-owych.

Skróty wprowadza się następująco:

```
@string{feb = {Luty}}
@string{feb = 02}
```

Przykład użycia znajduje się w rozdz. 4. W przykładzie tym ostatnie pole będzie równoważne

```
month = "8~02",
```

O użyciu skrótu świadczy jedynie brak nawiasów w zawartości pola nienumerycznego. Nie można więc użyć skrótu wewnątrz napisu, można jedynie złożyć pole z kilku napisów będących stałymi lub zmiennymi.

Skrót musi być zainicjalizowany *przed* użyciem; BibTEX sygnalizuje błąd, jeśli wartość skrótu jest pusta w momencie jego użycia. Można zmienić wartość skrótu, jak w powyższym przykładzie, również zdefiniowanego we wcześniej użytej bazie. Na przykład, baza bibliograficzna /texmf/bibtex/bib/amslatex/mrabbrev.bib z płyty TEXLive4 składa się wyłącznie z blisko 2000 skrótów nazw czasopism matematycznych. Wersja z roku 1997, znajdująca sie na serwerze GUST-u[1], zawiera już ponad 4000 skrótów.

Pewne skróty, głównie nazwy miesięcy, ale również kilku czasopism, są też zdefiniowane w stylach *.bst.

[1] Katalog macros/latex/required/amslatex/classes/, oraz ukryta na tejże płycie: systems/os2/emtextds/texmf.zip/bibtex/bib/

Skróty w skrótach Pewne zagnieżdżanie jest możliwe, a nawet sensowne. Na przykład tak:

```
@string{LNCS={Lect.\ Notes in Comp.\ Sci.}}
@string{LNCS1999=LNCS # {~1999}}
```

7.2. Odwołania w BibTEX-u

Czasami w bazie bibliograficznej występuje kilka pozycji o wspólnym rodowodzie, np. kilka artykułów z tego samego numeru czasopisma czy kilka tomów wielotomowego dzieła. Wówczas można podzielić informacje potrzebne do zebrania jednego rekordu na części. Na przykład

```
@InProceedings{bor:bibtex,
  crossref= {gust13},
  key     = {},
  author  = {Andrzej M. Borzyszkowski},
  title   = {\bibtex{} -- narzędzie do ..},
  }
@Proceedings{gust13,
  key     = {GUST13},
  booktitle= {Biuletyn {GUST}},
  title   = {Biuletyn {GUST}},
  year    = 1999,
  volume  = 13,
  publisher= gust,
  }
```

zamiast

```
@InProceedings{bor:bibtex,
  author  = {Andrzej M. Borzyszkowski},
  title   = {\bibtex{} -- narzędzie do ..},
  booktitle= {Biuletyn {GUST}},
  year    = 1999,
  volume  = 13,
  publisher= gust,
  }
```

Pierwszy rekord posiada niekompletne dane – w przykładzie brakuje nawet obowiązkowego pola `booktitle`. BibTEX uzupełni je danymi z drugiego rekordu. Powtórzenia nie są tutaj błędem, wartość pola z oryginalnego rekordu ma priorytet.

Rekord, do którego zwracamy się po dodatkowe informacje, musi być przetwarzany *po* rekordzie, który żądał tych informacji. A więc, albo musi się znajdować w tej samej bazie, ale później, albo musi być zawarty w bazie wymienionej później w instrukcji `\bibliography`.

Drugi rekord ma pełne prawo być osobną pozycją bibliograficzną. Fakt ten ma dwie konsekwencje. Po pierwsze, w drugim rekordzie, oprócz pola `booktitle` (na ogół ono właśnie jest polem brakującym w pierwszym rekordzie) należy umieścić pole `title` o identycznej lub zbliżonej zawartości. Niestety, nie można *zdefiniować* wartości pola `title` jako wartości pola `booktitle`. Po drugie, w rekordzie tym na ogół brakuje autora, powinien więc zawierać pole `key`. Ale pierwszy rekord nie powinien mieć niepustej zawartości pola `key`. W związku z tym użycie pola `crossref` idzie najczęściej w parze z deklaracją `key={}`.

Drugi rekord będzie osobną pozycją bibliograficzną, jeśli spełnia co najmniej jeden z warunków:

- jest osobno cytowany, choćby przez `\nocite`,
- co najmniej *dwie* cytowane pozycje odwołują się do niego.

W naszym przykładzie wygenerowana pozycja bibliograficzna `bor:bibtex` zamiast fragmentu

```
{\em Biuletyn {GUST}}, wolumen~13. Grupa
  U/zytkownik/ow Systemu \TeX, 1999.
```

zawierać będzie fragment

```
Patrz {\em Biuletyn {GUST}\/} \cite{gust13}.
```

a oprócz tego zostanie utworzona jeszcze jedna pozycja bibliograficzna,

```
\bibitem[GUS99]{gust13}
{\em Biuletyn {GUST}}, wolumen~13. Grupa
  U/zytkownik/ow Systemu \TeX, 1999.
```

Opcja `-m` wywołania BibTEX-a ustala ilukrotne odwołanie się do tego samego rekordu prowadzi do powyższych skutków.

Oczywiście, zawsze można wpisać wprost do bazy bibliograficznej

```
@InProceedings{bor:bibtex,
  author  = {Andrzej M. Borzyszkowski},
  title   = {\bibtex{} -- narzędzie do ..},
  booktitle= {\cite{gust13}},
   }
```

Wówczas naprzemienne przebiegi LATEX-a oraz BibTEX-a (ile, i dlaczego naprzemienne?) spowodują skutek zbliżony do poprzedniego. Pewną różnicą będzie to, że całym tytułem serii w pozycji `bor:bibtex` będzie `{\em \cite{gust13}}`, no bo taki dosłownie nadaliśmy jej `booktitle`. Zasadniczą różnicą będzie jednak fakt, że nawet jedno użycie `\cite{bor:bibtex}` spowoduje pojawienie się osobnej pozycji dla rekordu `gust13`, co można uważać za niepożądane.

7.3. Makra TEX-owe w BibTEX-u

Może się zdarzyć, iż chcemy użyć makr TEX-owych w bazie bibliograficznej. Co zrobić, by definicje tych makr zostały przeniesione do pliku z bibliografią? Wyjaśnia to przykład:

```
@preamble{ "\newcommand{\noopsort}[1]{} "
        # "\prefixing "}
```

Cały napis będący zawartością „rekordu” typu `preamble` zostanie przeniesiony do pliku z bibliografią. Znajdzie się on na początku tego pliku, choćby w bazie bibliograficznej był gdzie indziej. To, że napis został zdefiniowany jako konkatenacja napisów, nie ma żadnego znaczenia. Napis ten zostanie umieszczony w bibliografii nawet, jeśli żaden rekord z tej bazy nie został zacytowany. Jeśli używamy kilku baz, wniosą one swój wkład w kolejności ich wywołania.

I w tej części bazy BibTEX-owej można użyć mechanizmu skrótów: napis pozbawiony nawiasów klamrowych czy cudzysłowów będzie potraktowany jako skrót, napis w nawiasach będzie przeniesiony dosłownie.

Powyższe przykładowe makro LATEX-owe zostało zapożyczone z [Pat88a]; jaki mógłby być pożytek z tego i podobnych makr, można poczytać tamże.

8. Skąd brać style

Cztery style opisane w książce Lamporta [Lam86] są częścią dystrybucji TEX-a, istnieją też ich polskie odpowiedniki, patrz. rozdz. 6. Swoje style posiada AMS i wiele liczących się wydawców i czasopism. Zdarza się więc, że nie ma problemu z wyborem stylu – wyboru dokonał już kto inny. Jeśli jednak zechcemy wybrać gotowy styl, to często, wobec braku *jakiejkolwiek* dokumentacji, zadanie nie jest proste.

Można też pokusić się o własny styl. Ich język opisany jest w [Pat88b] – tylko przeczytać, przyswoić i zastosować. Komu niestraszna odwrotna notacja polska i maszyna stosowa, nie będzie miał problemów.

Dostępne jest jeszcze jedno rozwiązanie. Można mianowicie skorzystać z generatora stylów `custom-bib` autorstwa Patricka Daly. Jest to, formalnie rzecz biorąc, zestaw makr: na płycie TEXLive4 znajduje się w podkatalogu `/texmf/tex/latex`. Makra te nie służą jednak do składu tekstu, a do skorzystania z możliwości `docstrip`. Najpierw należy wywołać LATEX-a na pliku `makebst.tex`. Następnie odpowiedzieć na dziesiątki pytań dotyczących możliwych cech generowanego stylu BibTEX-owego. Powstaną wówczas dwa pliki, normalny log TEX-owy, zawierający nasze odpowiedzi oraz plik `*.dbj`. Teraz należy zadziałać LATEX-em na tymże pliku, i styl BibTEX-owy będzie gotowy. Plik `*.dbj` jest na tyle czytelny, że w przypadku małej modyfikacji stylu warto raczej zmodyfikować ten plik, niż przechodzić jeszcze raz przez procedurę odpowiedzi na 66 pytań. Style American Geophysical Union zamieszczone na płycie TEXLive4 zostały utworzone właśnie w ten sposób. Częścią pakietu `custom-bib` są pliki dotyczące języków. Plik polish.mbs można z łatwością napisać na kolanie. Niestety, uwagi o fleksji z rozdz. 6 są jak najbardziej aktualne, a wygenerowany styl nie będzie się odznaczał poprawną polszczyzną. Wspomnieć należy, że niektóre wygenerowane style pokażą swoje możliwości dopiero w połączeniu ze stylem LATEX-owym `natbib`, też autorstwa Patricka Daly i też łatwo dostępnym w wiadomych miejscach.

Rok 2000. Jest to hasło, bez którego nie może się dziś obejść żadna publikacja pretendująca do (a) bycia poważną oraz (b) poruszania tematyki komputerowej. A więc i my rozedrzyjmy szaty nad stylem BibTEX-owym stawiającym w etykiecie dwie tylko cyfry z roku publikacji. Kto chce, może poszperać w kwietniowym archiwum `gust-l` i dowiedzieć się, jak zmodyfikować style BibTEX-owe tak, by następny Bachotek nazywał się Bachotek 2000, a nie Bachotek ’00.

9. Zamiast zakończenia

Skąd brać bazy bibliograficzne? Pisaliśmy obszernie o naturalnym źródle baz bibliograficznych, tzn. o osobistej twórczości. Ale istnieje też inne źródło, interesujące przynajmniej dla pracowników wydziałów Computer Science uniwersytetów. Otóż wyszukiwarka internetowa `http://hypatia.dcs.qmw.ac.uk/` pozwala znaleźć dane w/w naukowców, również dane o ich publikacjach w formacie BibTEX-a – tylko brać i cytować.

BibTEX może być użyty również do implementacji prostej bazy danych nie mającej nic wspólnego z bibliografią, przykład znaduje się w zestawie `directory` na płycie TEXLive4. Jest tam kilka stylów BibTEX-owych: `address`, `birthday`, itd., oraz styl, tym razem LATEX-owy, do tworzenia wyciągów z tej bazy. Na płycie jest też kilka przykładowych baz dla tego zestawu.

Przykład ten jest skrajnym przypadkiem odejścia od standardowej struktury rekordów opisanej w [Lam86]. Inne ze stylów bibliograficznych raczej tylko rozszerzają zakres dostępnych pól.

Baza bibliograficzna jest plikiem tekstowym. A więc jedynym narzędziem naprawdę niezbędnym jest edytor.

Niektórzy jednak żądają więcej. Dla nich przeznaczonych jest kilka narzędzi ułatwiających obsługę baz bibliograficznych: wpisywanie rekordów, sprawdzanie poprawności bazy, być może sortowanie rekordów. Jednym z narzędzi jest wspomniany już pakiet AucTEX do Emacsa. Być może inne edytory też posiadają jego odpowiedniki. Istnieją też narzędzia wolnostojące. Przykładem niech będzie `bibtool` Gerda Neugebauera[2]. Również wspomniana wcześniej Hypatia poleca swoje narzędzia. Wydaje się, że sens używania osobnych narzędzi jest wątpliwy w przypadku niewielkich baz danych, tworzonych na własny użytek. Ale jeśli ktoś jest odpowiedzialny za utrzymanie porządku w większej bazie, to osobne narzędzia mogą mieć swój sens.

A tak na marginesie, BibTEX przetwarza teksty. Popularnym narzędziem do przetwarzania tekstów jest *Perl*. A więc nie dziwota, że wiele narzędzi około-BibTEX-owych jest po prostu skryptem perlowym, a i sam BibTEX też pewnie mógłby być.

Na zakończenie pozwolę sobie podziękować kolegom Ryśkowi Kubiakowi i Staszkowi Wawrykiewiczowi tudzież mojej Żonie za zachętę do napisania tego artykułu oraz staranne przejrzenie wstępnych wersji.

Pozycje cytowane i nie

[Bol96] Piotr Bolek. Bibliografia w LaTeXu – program bibtex, 1996. `http://www.ia.pw.edu.pl/Pl-iso/~bolek/bibtex/bibtex.html`.

[btxa] `texmf/bibtex/`.

[btxb] `texmf/doc/bibtex/`.

[Exa] `texmf/bibtex/bib/base/xampl.bib`.

[Lam86] Leslie Lamport. *LATEX: A Document Preparation System*. Addison-Wesley, 1986. Istnieje tłumaczenie polskie, Wyd. Ariel, tłum.: Piotr Wyrostek, 1992.

[Łoś00] Never Ever Łoś. No, nay, never, no more. Praca magisterska, Uniwersytet Powszechny, Styczeń 2000.

[Pat88a] Oren Patashnik. BIBTEXing. Documentation for general BIBTEX users, 8 Luty 1988.

[Pat88b] Oren Patashnik. Designing BIBTEX styles. The part of BIBTEX's documentation that's not meant for general users, 8 Luty 1988.

A. Przewodnik po implementacjach

Przeglądając płytę TEXLive4 oraz serwer GUST-u można zauważyć istnienie następujących implementacji

- `BibTeX, Version 0.99c (Web2C 7.3)`
  wydaje się, że jest to obecnie standardowa implementacja BibTEX-a, od MSDOS-a poczynając, a na różnych Uniksach kończąc.
- `BibTeX (bibtex32), Version 0.99c [4b]`
  autorstwa Eberharda Mattesa, przeznaczony dla OS/2 i MSDOS: `/systems/msdos/emtex/bibtex4b.zip`.
- `8-bit Big BibTeX version 0.99c`
  `Implementation:  GNU C for Unix`
  `Release version: 3.71 (18 Aug 1996)`
  autorstwa Kempsona i Aguilar-Sierra. Ta implementacja na płycie występuje wyłącznie w wersji dla Windows, na serwerze GUST-u leżą źródła[3], kompilacja pod Linux-a zajęła nie więcej czasu, niż trzeba by je ściągnąć i rozpakować.

Zaletą implementacji Web2C jest jej uniwersalność. W zasadzie można przystąpić do pracy z BibTEX-em natychmiast po zainstalowaniu TEX-a. Jednak sposób polonizacji tej implementacji budzi pewien niepokój. Łata Piotra Bolka, wspomniana w rozdz. 6, przeznaczona jest dla wersji Web2C 7.0. Obecnie obowiązuje wersja Web2C 7.3. A więc albo jesteśmy skazani na pogoń za najnowszą wersją, albo rezygnujemy z tej pogoni, pozostając przy wcześniejszej wersji BibTEX-a.

Implementacja Mattesa jest chyba standardem w świecie MSDOS-a, a na pewno OS/2. Także jest gotowa do natychmiastowego użycia po zainstalowaniu TEX-a dla tychże systemów. Wygląda jednak, że jest wroga każdemu językowi obcemu, nawet niemieckiemu!

Ostatnia z implementacji ma tę cechę, że nie jest związana z żadną dystrybucją TEX-a, nie zawiera też skompilowanej wersji pod Linux-a. Trzeba

[2] `/biblio/bibtex/utils/bibtool`

[3] `/biblio/bibtex/8-bit/bt371src.zip`

więc samemu kompilować wersję uniksową, a w każdym systemie zadbać o odpowiednie zmienne systemowe. Co gorsza, zmienne te wskazują na pojedyncze katalogi. Tymczasem zatrzęsienie dostępnych stylów bibliograficznych powoduje, że wolelibyśmy raczej przechowywać je w drzewku .../bst/. Implementacja ta ma jednak kilka zalet: posiada zdolność przetwarzania dużych baz, jest ośmiobitowa. Ale istotną jej zaletą jest odmienne podejście do problemów językowych. Mianowicie, korzysta ona z jeszcze jednego pliku danych, pliku `*.csf`, zawierającego informacje o sortowaniu oraz zamianie wielkości liter charakterystycznych dla danego języka. Polskie pliki dla stron kodowych 8859-2, 1250 oraz 852 dostępne są na serwerze GUST-u w `GUST/contrib/GUSTPROG/csf-pl.zip`.

B. Częste błędy

Istnieje cała gama możliwości popełnienia błędu, zarówno podczas tworzenia bazy bibliograficznej, jak i podczas jej późniejszego użycia. Omówimy tu tylko niektóre z nich.

B.1. Błędy w bazie

Problemy z wielkością liter. LaTeX rozróżnia małe i wielkie litery w etykietach, `\cite{gust}` i `\cite{GUST}` odwołują się do różnych pozycji. Jednak BibTeX nie chce tego uznać:

```
Case mismatch error between cite keys GUST
   and gust---line 60 of file manual.aux
 : \citation{GUST
 :              }
I'm skipping whatever remains of this
   command
```

Podwójne wystąpienie tej samej etykiety. Niezależnie od przyczyny powtórzenia, *pierwsze* wystąpienie etykiety będzie użyte do stworzenia `\bibitem`-u, a następne sygnalizowane będą jako błąd:

```
Database file #1: manual.bib
Database file #2: btxdoc.bib
Repeated entry---line 58 of file btxdoc.bib
 : @book{latex
 :           ,
I'm skipping whatever remains of this entry
```

Przyczyną może być zmiana etykiety w dawniej stworzonej bazie, np. z powodu zmiany roku produkcji pozycji bibliograficznej. Trzeba tu zachować dużą ostrożność i nie wprowadzać poprawek pochopnie. Inna przyczyna została omówiona przed chwilą – pomieszanie wielkości liter. Może też się zdarzyć, że celowo zmieniamy w naszej bazie rekord już istniejący w bazie ogólnodostępnej. Wówczas bazy ze zmienionym rekordem musimy użyć wcześniej, musimy też zamknąć oczy i uszy na sygnały błędów – praktyka godna nagany.

Specjalne traktowanie liczb. Tekst pola w rekordzie w zasadzie musi być ujęty w nawiasy, jednak zwolnione z tego obowiązku są pojedyncze liczby. Przyzwyczajenie spowodować może następujący scenariusz: systematycznie piszemy np.

```
number = 64
```

Przychodzi chwila, gdy trafi się podwójny numer,

```
number = 65/66
```

i błąd jest gotowy. Na szczęście jest natychmiast sygnalizowany.

Brak spacji. Należy pamiętać, iż tylko spacje zawarte w napisach przechodzą do pliku bibliograficznego. Jeśli używamy napisów łączonych, to mogą nas spotkać niespodzianki. Można zdefiniować skrót

```
@string{LNCS={Lect.\ Notes in Comp.\ Sci.}}
```

i użyć go dalej:

```
series = lncs # 1999,
```

nie będzie jednak spacji pomiędzy napisem a liczbą. Oczywiście

```
series = lncs # ~1999,
```

jest nielegalne, bo napis inny niż liczba musi być otoczony nawiasami.

Brak pola obowiązkowego. Wówczas staje się ono puste

```
Database file #1: manual.bib
Warning--empty author in gust-13
```

Takie zachowanie może być celowe, gdy np. chcemy umieścić w bibliografii 13. numer biuletynu GUST-u, a z braku rekordów typu „journal” umyśliliśmy użyć rekordu typu `article` bez podawania konkretnego autora.

Konsekwencje mogą być bardziej dalekosiężne, np.:

```
Database file #1: manual.bib
Warning--to sort, need author or key
  in gust-13
Warning--empty author in gust-13
```

Pole `key` służy pomocą właśnie do ustalenia etykiety, gdy sam rekord jest zbyt ubogi.

Skróty BibTEX-owe. Zupełnie jak w TEX-u.

Brak definicji powoduje:

```
Database file #1: manual.bib
Warning--string name "gust" is undefined
--line 35 of file manual.bib
Warning--empty publisher in gust-13
```

Najpierw skrót `gust` potraktowany został jako pusty napis, potem okazało się, że jakieś pole jest puste, a nie powinno. Jeszcze raz: skrót może być zdefiniowany nie tylko w bazie bibliograficznej, ale i w stylu, a styl rzeczą zmienną jest.

Podwójnie zdefiniowane skróty: późniejsza definicja zwycięża bez ostrzeżenia, mieliśmy już tego przykład wcześniej, patrz rozdz. 7.1.

W przeciwieństwie do TEX-a, użycie skrótu w swojej własnej definicji nie powoduje nieskończonej pętli, jedynie ostrzeżenie i niezdefiniowanie tegoż skrótu:

```
Database file #1: manual.bib
Warning--string name "feb" is used in its
  own definition--line 3 of file manual.bib
```

Przecinki. Całkiem jak w językach programowania, brak przecinka po polu jest sygnalizowany dopiero przed następnym polem. Ciekawostką jest, iż błąd jest sygnalizowany również dla rekordów, których wcale nie używamy:

```
I was expecting a `,' or a `}'---line 54
  of file btxdoc.bib
 :
 :    title= "The {{\TeX}book}",
(Error may have been on previous line)
I'm skipping whatever remains of this entry
```

Nieporozumienie z cudzysłowem. Jeśli używamy cudzysłowów jako nawiasów dla zawartości pól, to czeka nas niespodzianka. Rekord zawierający fragment:

```
@Article{mf,
   author = "Martin-L\"of",
```

nie może zostać niezauważony. Radą jest używanie nawiasów klamrowych, przy okazji łatwiej się sprawdza parowanie, lub pisanie:

```
@Article{mf,
   author = "Martin-L{\"o}f",
```

Nie jest tylko jasne co robić, gdy ktoś zechce mieć w tytule nawiasy klamrowe, szczególnie nie do pary.

Niezbalansowane nawiasy. Konsekwencje są podobne jak w TEX-u, błąd ujawnia się gdzie indziej niż powstał i powoduje trochę zamieszania.

Inne błędy syntaktyczne. BibTEX może sygnalizować błędy także wówczas, gdy nie dotyczą potrzebnych nam rekordów. Po prostu baza bibliograficzna musi być syntaktycznie poprawna.

B.2. Błędy użycia

Argument dla BibTEX-a podany z rozszerzeniem. Skutek zależy mocno od implementacji. Może się zdarzyć, że wskutek wykonania polecenia

```
  bibtex manual.aux
```

BibTEX uprze się czytać z pliku `manual.aux.aux`, którego oczywiście nie ma. Może czytać z pliku `manual.aux`, a jedynie zapisywać swoje wyniki do pliku `manual.aux.bbl`. Może odmówić działania, tłumacząc się wyraźnie czemu. Może nawet – dotyczy to tylko starszej wersji i w kontekście systemu plików FAT – zniszczyć zawartość `manual.aux`.

Wiele poleceń `\bibliography`. Tylko pierwsze użycie `\bibliography` będzie miało pożądany skutek, dalsze będą sygnalizowane jako błąd:

```
The top-level auxiliary file: manual.aux
Illegal, another \bibdata command---line
  62 of file manual.aux
 : \bibdata
 :          {btxdoc}
I'm skipping whatever remains of this
  command
```

W szczególności błędem jest wymienianie różnych baz za pomocą osobnych instrukcji, jak też powtórzenie instrukcji `\bibliographystyle`.

Pusty plik bibliograficzny. BibTEX ma prawo wyprodukować pusty plik bibliograficzny:

```
\begin{thebibliography}{}

\end{thebibliography}
```

Wówczas LATEX będzie bezsensownie narzekać, nim złoży pusty rozdział z bibliografią. Jedną z przyczyn stworzenia pustego pliku może być użycie BibTEX-a na wyrost, inną mogą być różne błędy.

Brak bazy bibliograficznej.

```
I~couldn't open database file manual.bib
---line 61 of file manual.aux
 : \bibdata{manual
 :                ,btxdoc}
I'm skipping whatever remains of this
  command
The style file: plalpha.bst
I~found no database files---while reading
  file manual.aux
```

Przyczyną jest raczej reorganizacja (zmiana katalogu, inna wersja BibTEX-a, brak deklaracji zmiennych systemowych) niż całkowite zniknięcie bazy. Skutek może być dalekosiężny, np. nie zostaną zauważone dalsze, jak najbardziej istniejące bazy, i w końcu może zostać utworzony pusty plik bibliograficzny. Jeśli bazy naprawdę nie ma i nie jest potrzebna, to należy starannie usunąć odwołania do niej.

Brak rekordu.

```
Warning--I didn't find a~database entry
  for "nnn"
```

Skutek jest taki sam jak cytowanie nieistniejących pozycji bibliograficznych: w trakcie kompilacji wystąpią uporczywe ostrzeżenia, a w złożonym tekście pojawią się podwójne znaki zapytania w nawiasach kwadratowych. Chyba, żeby brakujący rekord był wspomniany jedynie jako `\nocite{nnn}`, wówczas wszelki ślad po nim zaginie.

Przyczyną może być znowu zmiana etykiety w dawniej utworzonej bazie.

C. Kilka przypadkowych rad

Wszystkie pozycje z bazy `manual.bib` można wydrukować używając instrukcji

```
\nocite{*} \bibliography{manual}
```

Tekst w bazie bibliograficznej nie należący do żadnego rekordu jest ignorowany. Nie ma więc potrzeby wykonywania specjalnych zabiegów, gdy zależy nam na umieszczeniu komentarza w bazie danych – po prostu go umieszczamy, por. bazy na płycie TEXLive4. Warunek – za żadne skarby nie może znaleźć się tam znak `@`. Można stosować znak % na początku wiersza – byśmy i my wiedzieli, że to tylko komentarz. Nie jest jednak prawdą, jakoby sam procent był znakiem początku komentarza.

Jeśli chcemy umieścić w bazie tymczasowy rekord – taki, który ma być niewidoczny dla BibTEX-a – to wystarczy po prostu usunąć znak `@` z jego początku.

Można wprowadzać do rekordów pola `annote`, `abstract`, itp., które będą ignorowane przez codziennie używane style, ale które pozwolą czasami przygotować np. bibliografię z naszymi komentarzami. Można też umieszczać pola, których nie mamy zamiaru nigdy ujawnić, a służą wyłącznie do obsługi samej bazy danych, np. datę umieszczenia pozycji w bazie albo źródło pochodzenia informacji o pozycji.

Nie jest jednak prawdą, że miejscem na nasze notatki jest pole `note`. Wszystkie standardowe style BibTEX-owe kopiują zawartość pola `note` do pliku z bibliografią.

Pole o zawartości

```
author = {Andrzej M. Borzyszkowski},
```

daje w bibliografii wynik

```
Andrzej~M. Borzyszkowski.
```

Tylda między pierwszym a drugim imieniem jest mało potrzebna. Ale kropka po inicjale imienia wnosi negatywny wkład do wyglądu całości, przynajmniej w wersji angielskojęzycznej (brak spacjowania francuskiego). Warto więc umieszczać w-tył-ciach po kropce kończącej inicjał; BibTEX przeniesie go do tekstu bibliografii.

Obowiązkiem BibTEX-a jest wykrycie najszerszej etykiety użytej w bibliografii. Np. fragment

```
\begin{thebibliography}{Pat88b}
```

świadczy o uznaniu `Pat88b` za taką etykietę. Ale przecież BibTEX nie ma szans wiedzieć, jakimi czcionkami będzie dokonywany skład bibliografii i czy przypadkiem `Lam86` nie zajmie więcej miejsca. Może tu zajść konieczność ingerencji w wygenerowany plik.

Nie warto robić poprawek w pliku `manual.bbl`. Następny przebieg BibTEX-a, choćby przypadkowy, spowoduje zniknięcie poprawek. Jeśli naprawdę chcemy zaingerować, lepiej zmienić nazwę pliku i zadbać, by był czytany przez LATEX-a. A na pewno trzeba tak postąpić, gdy tworzymy kilka plików bibliograficznych.

◇ Andrzej M. Borzyszkowski
a.borzyszkowski@ipipan.gda.pl