Wersja z 25 grudnia 2003

# Anotacja zewnętrzna: wpływ architektury korpusu IPI PAN na efektywność jego tworzenia oraz wykorzystania*

Piotr Bański
Instytut Anglistyki,
Uniwersytet Warszawski
bansp@venus.ci.uw.edu.pl

## 1. Wstęp

Celem niniejszej pracy jest przedstawienie architektury korpusu IPI PAN oraz zalet opisanego systemu. Ogólne informacje na temat korpusu można znaleźć w artykule Przepiórkowskiego i in. (2003), tutaj więc skupimy się jedynie na sposobie opisu danych tekstowych oraz korzyściach, jakie taki rodzaj opisu stwarza. Zanim jednak przejdziemy do szczegółów, wyjaśnimy powody użycia terminu *anotacja*, a następnie naszkicujemy rozumienie takiego pojęcia efektywności korpusu, jakie nas tu będzie interesowało. Punkt 2 niniejszej pracy opisuje ogólną strukturę korpusu IPI PAN w obecnej fazie jego rozwoju[1], punkt 3 traktuje o koncepcji anotacji zewnętrznej, a punkt 4 pokazuje jej zalety, m.in. w odniesieniu do omawianego korpusu.

### 1.1. Pojęcie anotacji

Termin *anotacja* budzi czasem kontrowersje jako neologizm. Jest on jednak terminem o znaczeniu dość łatwym do scharakteryzowania, a ponadto przydatnym przy opisie problemów związanych z architekturą oraz typem danych wyróżnianych i manipulowanych w językoznawstwie korpusowym. Wydaje się, iż potencjalne ujemne koszta estetyczne przyjęcia tego terminu zrównoważone zostaną jego przydatnością, wynikającą także z braku zastanego rodzimego odpowiednika, którego można by użyć w odpowiednio wąskim rozumieniu, a także z konieczności oddzielenia terminu *znakowanie* jako odnoszącego się do innego zakresu znaczeń.

*Słownik naukowo-techniczny angielsko-polski* WNT z roku 1990 proponuje terminy *adnotacja* oraz *komentarz* jako tłumaczenia angielskiego *annotation*. Terminy te nie obejmują interpretacji użytej w tytule niniejszej pracy, być może ze względu na datę kompilacji i publikacji *Słownika*.

Za przykładem m.in. publikacji Ide i Brew (2000) oraz McEnery i Wilson (2001), w niniejszej pracy przyjmiemy podział sposobów opisu korpusów w sensie ogólnym na anotację (ang. *annotation*) oraz znakowanie (ang. *mark-up*, *encoding*).[2] To pierwsze odnosi się do

* Autor pragnie wyrazić wdzięczność Beacie Chachulskiej — za cierpliwość, oraz Adamowi Przepiórkowskiemu — za wyrozumiałość i cenne uwagi.

Niniejsza wersja artykułu różni się nieznacznie od wersji opublikowanej w *Polonikach XII*. W przypadku cytowania bardzo proszę o odwoływanie się do wersji oficjalnej.

[1] Pragniemy zastrzec się, iż architektura korpusu może jeszcze ulec pewnym modyfikacjom.

[2] Dwuznaczność terminu *annotation* jest wyraźnie widoczna np. w opisie Corpus Encoding Standard (<http://www.cs.vassar.edu/CES/>). W rozdziale 4.5.8.7. (CES1-4.5.html#ToC18) użyto tego terminu w odniesieniu do *komentarzy* anotatorów, podczas gdy wszystkie inne jego wystąpienia odnoszą się do tego, co nazywamy tu anotacją. Termin *encoding* (którego używają np. McEnery i Wilson) jest bardziej ogólny i wieloznaczny niż *mark-up*, co ma wyraz także w tym, iż używa się go w znaczeniu np. kodowania znaków (*character encoding*). W niniejszej pracy ograniczymy się do zakresu pojęcia *znakowanie* wyrażonego angielskim *mark-up*, które (przy nieznacznych uproszczeniach) interpretowane jest zwykle jako wyodrębnianie fragmentów tekstu za pomocą umownych oznaczeń czysto tekstowych, w sposób niezależny sprzętowo —

opisu danych, a także do relacji pomiędzy danymi a ich opisem (lub pomiędzy poszczególnymi poziomami opisu — do tego wracamy w pkt. 3). Drugie pojęcie określa sposób doboru znaczników oraz ich fizycznego umieszczenia w strumieniu danych tak, aby owe abstrakcyjne relacje ukonkretnić. Jest to rozróżnienie analogiczne do rozróżnienia pomiędzy abstrakcyjnym algorytmem z jednej strony, a jego ukonkretnieniem w formie programu zapisanego w takim czy innym języku programowania. Naturalnie, nie każdy język programowania jest jednakowo przydatny do oddania niuansów danego algorytmu. Podobnie z systemem znakowania — w naszym przypadku jest to system zwany XCES, będący wersją standardu CES (Corpus Encoding Standard) zdefiniowaną w języku XML (Bray i in. 2000), uznawanym za faktyczny standard znakowania zbiorów danych, a więc w szczególności korpusów językowych.[3]

Potencjalnie nieco węższa interpretacja terminu *anotacja*, jako 'lingwistyczne znakowanie korpusu', przyjęta jest w pracy Przepiórkowski i in. (2003).[4] Jest to zapewne rezultatem specjalnego statusu terminów w rodzaju *POS (Part-of-Speech) annotation* czy też (*morpho*)*syntactic annotation*, odnoszących się do sposobów (morfo)syntaktycznego opisu form wyrazowych. Jeśli nie będzie to dodatkowo zaznaczone, w niniejszej pracy będziemy traktować termin *anotacja* w najszerszym ze znaczeń przytoczonych powyżej, czyli za Ide i Brew (2000).

Należy dodać jeszcze, iż nie uważamy, aby okazjonalne zamienne użycie terminów *anotacja* i *znakowanie*, spotykane czasem w tekstach zarówno anglo- jak i polskojęzycznych, musiało być uznane za błąd: znakowanie to fizyczna manifestacja anotacji, a dość powszechną i — wydaje się — naturalną praktyką jest utożsamianie znaku i jego desygnatu, jeśli względy teoretyczne nie nakazują kategorycznego rozdzielenia tych dwóch obiektów. Ważne jest, aby mieć świadomość ich odrębności.

### 1.2. Pojęcie efektywności

Efektywność rozumiemy tu jako minimalizację nakładu pracy oraz środków pieniężnych przy jednoczesnej maksymalizacji korzyści płynących z wykorzystania korpusu.

Minimalizacja nakładu pracy nie oznacza po prostu zautomatyzowania całego procesu tworzenia korpusu — jak zobaczymy, część zadań wykonywanych przy tworzeniu korpusu IPI PAN może wykonać tylko człowiek — bardziej lub mniej wykwalifikowany anotator. Sztuka polega tu raczej na podziale zadań na (niemal) w pełni zautomatyzowane, wymagające anotatora niewykwalifikowanego oraz te, które muszą być wykonane przez anotatorów wykwalifikowanych (w naszym przypadku językoznawców). Sprawia to, iż możliwe jest dokładne wyznaczenie zakresu pracy anotatora wykwalifikowanego, a zarazem, z uwagi na to, że praca anotatora z natury trwa o wiele dłużej niż znakowanie automatyczne, możliwe jest także przesunięcie danego zadania w taki sposób, aby nie spowalniało ono procesu produkcji podstawowych części korpusu. Wracamy do tego zagadnienia w punkcie 4.1.

---

stosuje się to zarówno do znaczników w stylu SGML/XML, jak i formatu COCOA, czy też np. tekstu podzielonego na kolumny o stałej szerokości.

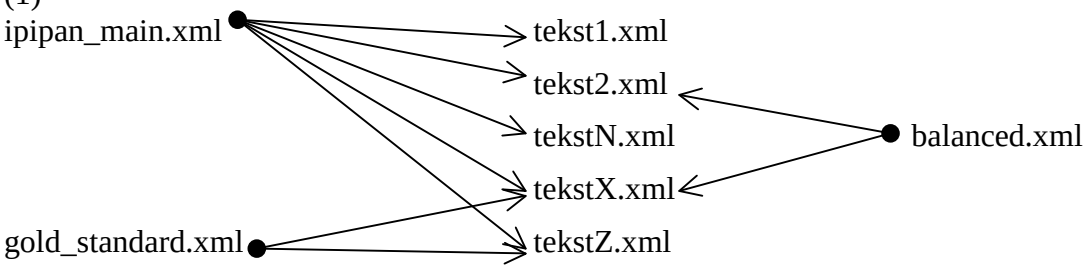
[3] XCES (<http://www.cs.vassar.edu/~ide/xces-0.2/>) jest aplikacją XML-a, podobnie jak HTML, język opisu stron WWW, jest aplikacją języka SGML (standardu ISO 8879). SGML oraz XML same w sobie nie są schematami opisu, a jedynie językami umożliwiającymi definiowanie takich schematów. Schemat TEI (<http://www.tei-c.org/>), na którym zostało oparte CES (z którego z kolei wyrosło XCES) był początkowo aplikacją SGML-a, a ostatnio (prace zakończono w roku 2002) został przedefiniowany jako aplikacja XML-a.

[4] Piszemy „potencjalnie węższa", gdyż zależy to np. od tego, czy znakowanie granic zdań i wszystkich wystąpień tekstu wyróżnionego np. kursywą lub tłustym drukiem będzie uznane za znakowanie lingwistyczne. Wydaje się, że taka praktyka jest możliwa, aczkolwiek niekonieczna.

## 2. Ogólny kształt korpusu

Zakładamy, iż korpus IPI PAN składać się będzie z trzech hierarchii, jak pokazano poniżej. Korzeniem (ang. *root*) drzewa plików składających się na korpus właściwy jest plik ipipan_main.xml. Plik balanced.xml odgrywa tę samą rolę wobec podkorpusu zrównoważonego, a plik gold_standard.xml — wobec podkorpusu sprawdzanego ręcznie. Taka struktura jest wymuszona właściwościami języka XML oraz tym, że większość znaczników w tekstach korpusu posiada atrybuty ID, których wartościami są jednoznaczne identyfikatory danego elementu. Jak widać poniżej, te trzy hierarchie nakładają się na siebie. Gdyby były one połączone w jedną hierarchię (tj. gdyby pliki balanced.xml i gold_standard.xml były częściami drzewa plików, którego korzeniem jest plik ipipan_main.xml), oznaczałoby to niedozwolone w XML-u powtarzanie się atrybutów ID w obrębie korpusu i niemożliwość walidacji całości bez uprzednich ‘zabiegów kosmetycznych’, których chcemy uniknąć.[5]

(1)




Pliki wchodzące w skład podkorpusu zrównoważonego dobierane są pod kątem kategorii tekstu, jaki zawierają i dlatego najłatwiej będzie je po prostu wyliczyć w pliku głównym tego korpusu, tj. w pliku balanced.xml.[6] Jeśli chodzi o podkorpus sprawdzany ręcznie, to optymalny mechanizm identyfikacji plików włączonych w ten korpus jest odmienny niż w korpusie zrównoważonym. W tym drugim, fakt przynależności danego pliku do korpusu nie jest zaznaczony wewnątrz nagłówka tego pliku — jak wspomnieliśmy, kwestia przynależności do korpusu zrównoważonego wynika z przyjętych statystyk mających zapewnić mu reprezentatywność. Inaczej jest z tekstami weryfikowanymi ręcznie — fakt ręcznej weryfikacji oznaczany jest m.in. w nagłówku danego pliku, i dlatego metodą najmniej podatną na błędy jest wyszukanie takich plików, najlepiej za pomocą mechanizmów transformacji XSL (XSLT, <http://www.w3.org/TR/xslt>), i umieszczenie rezultatów wyszukiwania w pliku głównym gold_standard.xml.

Fragment struktury plików korpusu IPI PAN pokazany jest poniżej. Katalog główny korpusu dzieli się na podkatalogi według przyjętej taksonomii tekstów. Każdy z tych podkatalogów zawiera kolejne zestawy podkatalogów, aż do poziomu najniższego (2c).

[5] Zapewne już niedługo taka struktura nie będzie czymś logicznie koniecznym, zważywszy na to, że adresowanie za pomocą elementów ID będzie można wyeliminować na rzecz adresowania wyłącznie za pomocą mechanizmów XPath oraz XPointer. Użycie elementów ID wynika w tym momencie ze względów praktycznych — jest to mechanizm sprawdzony i zwarty, podczas gdy język XPointer wciąż jest w fazie ewolucji. Z podobnego powodu, po początkowej fascynacji mechanizmem XML Schema, wprowadzanym do nowej wersji XCES, zdecydowałem się wrócić do tradycyjnego mechanizmu DTD, ponieważ jest on bardzo stabilny i zupełnie wystarczający do opisu struktury korpusu IPI PAN, a ponadto umożliwia on wykorzystanie mechanizmu ‘całostek’ (ang. *entities*) do włączania poszczególnych plików składowych w całość hierarchii korpusu. Mechanizm XInclude, który ma przejąć większość zastosowań całostek, nie jest jak na razie szeroko stosowany w narzędziach do obróbki XML-a, podczas gdy każdy parser XML-a rozumie format DTD (będący częścią definicji XML 1.0) i radzi sobie z całostkami.

[6] Jest to pewne uproszczenie: odpowiednie pliki ‘wyliczane’ są pośrednio, za pomocą mechanizmu całostek.

(2) a. Częściowa zawartość katalogu głównego korpusu IPI PAN

| | | |
|---|---|---|
| drama/ | — | katalogi poszczególnych podkorpusów |
| fiction/ | | |
| newspapers/ | | |
| (...) | | |
| balanced.xml | | |
| gold_standard.xml | | |
| ipipan_main.xml | | |

b. Częściowa zawartość katalogu newspapers/

| | | |
|---|---|---|
| n1/ | — | podkatalogi wewnątrz podkorpusu |
| n2/ | | (struktura uproszczona) |
| (...) | | |
| newspapers.xml | | |

c. Zawartość katalogu n1/

| | | |
|---|---|---|
| header.xml | — | nagłówek |
| lex.xml | — | anotacja morfosyntaktyczna |
| sent.xml | — | segmentacja zdaniowa, uszczegółowienia |
| text.xml | — | tekst właściwy |

Z części (2a) i (2b) pominięto wszelkiego rodzaju pliki oraz katalogi pomocnicze.[7] Nie będziemy się w tym momencie zajmować strukturą katalogów na dole hierarchii, w (2c), jednakże pokazana jest ona dla dopełnienia obrazu całości oraz po to, aby podkreślić rolę i miejsce nagłówków w korpusie.

Poniższe przykłady pokazują najogólniejszą strukturę głównego pliku korpusu ipipan_main.xml (3a), głównego pliku podkorpusu, np. newspapers.xml (3b) oraz pliku text.xml, np. w katalogu newspapers/n1/ (3c).

(3) a.
```
<cesCorpus>
  <cesHeader> ... </cesHeader>
  <cesCorpus> ... </cesCorpus>   (element powtarzający się, zob. 3b)
</cesCorpus>
```

b.
```
<cesCorpus>
  <cesHeader> ... </cesHeader>
  <cesDoc>  ... </cesDoc>        (element powtarzający się, zob. 3c)
</cesCorpus>
```

c.
```
<cesDoc>
  <xcesHeader/>        (dowiązanie do pliku header.xml z 2c)
  <text> ... </text>
</cesDoc>
```

[7] Są to pliki zawierające różnego rodzaju definicje i wywołania dwóch rodzajów całostek, za pomocą których poszczególne pliki korpusu łączone są w jedną hierarchię.

Każdy dokument korpusu składa się z dwóch podstawowych części. Jedną z tych części jest nagłówek, a drugą (w zależności od miejsca danego dokumentu w hierarchii) zbiór podkorpusów (3a), zbiór dokumentów tekstowych najniższego szczebla (3b), lub wreszcie pojedynczy tekst (3c).

Zauważmy, że oddzielny plik nagłówka pojawia się dopiero na najniższym poziomie struktury — nagłówki głównego pliku korpusu oraz nagłówki podkorpusów nie są wydzielane, ponieważ nie zachodzi taka potrzeba — są one modyfikowane tylko sporadycznie, a ponadto modyfikowanie ich nie niesie za sobą niebezpieczeństwa przypadkowej utraty danych: z ich poziomu nie ma fizycznego dostępu do tekstów korpusu. Niebezpieczeństwo takie pojawia się bardziej realnie w przypadku nagłówków plików text.xml, tym bardziej, że nagłówki tych plików modyfikowane są z założenia częściej niż inne nagłówki. Jest to spowodowane tym, że zawartość nagłówka musi odzwierciedlać wszelkie zmiany dokonywane nie tylko w samym pliku text.xml, ale także w katalogu zawierającym ten plik. Tak więc nagłówek modyfikowany jest po raz pierwszy przy tworzeniu pliku text.xml (ta faza jest z reguły wykonywana automatycznie), a następnie przy każdej korekcie lub uzupełnieniu jego zawartości, a także wtedy, gdy tworzone lub modyfikowane są pliki dodatkowe, takie jak sent.xml czy lex.xml (zob. 2c). Jak zobaczymy, niektóre z tych plików mogą (a nawet czasem powinny) być modyfikowane ręcznie, co zwiększałoby możliwość przypadkowej ingerencji w dane zawarte w pliku text.xml, gdyby nagłówek był fizycznie częścią tego pliku. Ponadto, jako najcenniejsza część korpusu, pliki text.xml są zachowywane jako pliki tylko do odczytu, i dzięki temu, że nagłówki tych plików są fizycznie wydzielone, przy ich modyfikacji nie zachodzi potrzeba usuwania blokady zapisu dla plików text.xml. Taką niezależność nagłówków w połączeniu z jednoczesnym zabezpieczeniem tekstów traktujemy jako jeszcze jeden wyraz efektywności przyjętej struktury korpusu, rozumianej chociażby w sposób trywialny: jako oszczędność pracy i środków finansowych, które byłyby konieczne w przypadku utraty danych.

## 3. Koncepcja anotacji zewnętrznej

W tej części wspominamy najpierw o mechanizmie dowiązań, który wykorzystany jest do implementacji anotacji zewnętrznej (3.1), następnie w punkcie 3.2 omawiamy ‘warstwową’ strukturę plików na najniższym poziomie korpusu IPI PAN (por. 2c), aby na tej podstawie omówić szczegółowiej samo pojęcie anotacji zewnętrznej (3.3).

### 3.1. Dowiązania

Podobnie jak SGML, XML jest metajęzykiem służącym do konstrukcji konkretnych języków znaczników, zwanych aplikacjami danego metajęzyka. Aplikacjami SGML-a były m.in. HTML oraz TEI do wersji P3. Aplikacjami XML-a są np. XHTML, TEI P4 oraz XCES.

Aplikacją jednej z wczesnych wersji TEI jest schemat CES, stworzony w ramach projektów EAGLES, MULTEXT oraz MULTEXT-EAST, specjalnie na potrzeby znakowania korpusów. Każdy tekst oznakowany tym schematem spełnia także warunki nałożone na teksty oznakowane za pomocą TEI, a przez to jest też jednocześnie pełnoprawną aplikacją SGML-a.

Zależność pomiędzy XCES a TEI P4 jest nieco inna, gdyż translacja CES na XCES przebiegała mniej więcej równolegle z translacją TEI P3 na TEI P4. O ile więc zarówno XCES jak i TEI P4 są aplikacjami XML, o tyle relacja pomiędzy nimi nie jest już relacją

bezpośrednią, choć pod bardzo wieloma względami XCES może wciąż być uważany za ukonkretnienie TEI P4.[8]

CES przejęło od TEI koncepcję dowiązań (ang. *links*), które w swej podstawowej formie znane są powszechnie z języka HTML, używanego do tworzenia stron WWW. Dowiązania w HTML-u służą najczęściej do nawigacji wewnątrz dokumentów lub pomiędzy nimi: 'klikając' w odpowiednim miejscu dokumentu hipertekstowego uruchamiamy takie dowiązanie, które prowadzi nas ze źródła (odpowiednio oznakowanego tekstu) do celu (innego dokumentu lub jego fragmentu, lub też serwisu sieciowego w rodzaju FTP, telnetu lub poczty elektronicznej). Twórcy TEI zaproponowali, aby użyć dowiązań do znakowania np. przypisów lub innych odnośników do partii tekstu, w tym samym lub w różnych dokumentach. Można też w ten sposób łączyć w całość elementy pochodzące z różnych miejsc drzewa SGML (na przykład wszystkie akapity lub zdania, w których autor odnosi się do określonej postaci historycznej lub wszystkie kwestie wygłoszone przez wybraną postać dramatu itp.) Zaproponowano także sposób, w jaki analogiczne elementy z różnych tekstów mogą być ze sobą uzgadniane — w ten sposób można np. łączyć tekst oryginalny z jego tłumaczeniem w innym języku tak, aby pokazać wzajemnie odpowiadające sobie zdania lub słowa.[9]

Jedną z dodatkowych propozycji CES było użycie dowiązań do łączenia pliku zawierającego anotację morfosyntaktyczną z plikiem anotowanym strukturalnie. XCES przejęło i uogólniło ten mechanizm na inne rodzaje anotacji. Jako aplikacja XML-a, XCES ma dostęp do wszystkich możliwości, jakie posiada XML, w tym także możliwości tworzenia przeróżnego rodzaju dowiązań. Dowiązania w XML-u definiowane są przez jego podsystem zwany XLink, który dodatkowo rozszerza ich właściwości tak, że możliwe jest zbudowanie dowiązania o dwóch lub więcej członach, z których żaden nie może być nazwany początkowym lub docelowym.[10] Takie połączenia przydatne są przy tworzeniu korpusów równoległych, których zadaniem jest połączenie kolejnych jednostek tekstowych (np. zdań lub słów) w oryginale i jego tłumaczeniu. Jak zobaczymy w punkcie 4, architektura korpusu IPI PAN umożliwia *bezpośrednie* wykorzystanie jego tekstów do konstrukcji korpusów równoległych lub porównawczych, bez najmniejszych ich modyfikacji. Jest to jeden z pierwszych korpusów na świecie skonstruowanych według takich zasad.

---

[8] Kolejnym czynnikiem rodzielającym te dwa systemy jest wybór języka opisu struktury i zawartości dokumentów XML, jako następcy DTD: kreatorzy XCES optują za XML Schema, podczas gdy zespół tworzący nową wersję TEI używa języka RELAX NG. Ten rozziew nie będzie być może tak poważny, na jaki wygląda, jeśli prace jednego z twórców RELAX NG, Jamesa Clarka, nad stworzeniem metod translacji pomiędzy obydwoma systemami, uwieńczone zostaną sukcesem, jak sugeruje strona <http://www.thaiopensource.com/relaxng/trang.html>.

[9] Jako przykład takiego korpusu może służyć korpus równoległy będący ukoronowaniem projektów MULTEXT-EAST oraz CONCEDE (<http://nl.ijs.si/ME/V2/>), w ramach których rozwijane było CES. Jest to korpus złożony z elektronicznej wersji książki *1984* George'a Orwella oznakowanej strukturalnie i morfosyntaktycznie, oraz z jej podobnie oznakowanych przekładów na sześć języków (bułgarski, czeski, estoński, rumuński, słoweński oraz węgierski). W korpusie tym istnieją dwa rodzaje uzgodnień: dwustronne (pomiędzy wersją angielską i każdym z tłumaczeń) oraz siedmiostronne, gdzie każda wersja jest równorzędnym węzłem sieci tłumaczeń. Struktura tego korpusu podkreśla także bliski związek pomiędzy TEI a CES: teksty główne, zawierające opis morfosyntaktyczny, znakowane są w systemie TEI, lecz dokumenty zewnętrzne uzgadniające przekłady są dokumentami CES.

Przykładem innego rodzaju korpusu równoległego jest minikorpus dostępny na stronach XCES. W tym korpusie, tekst angielski jest uzgodniony z tekstem francuskim na poziomie poszczególnych wyrazów.

[10] Na liście dyskusyjnej XML-DEV (http://lists.xml.org/archives/xml-dev/), która dała początek wielu projektom związanym z rozwojem XML-a, przewija się opinia, że XLink jest specyfikacją martwą i że należy stworzyć inny system, który by ją zastąpił. Będzie to miało pewne, trudne w tej chwili do ustalenia, konsekwencje dla korpusów używających systemu XCES. Dla prostych dowiązań używanych w korpusie IPI PAN wpływ takiej zmiany będzie znikomy.

### 3.2. Struktura plików na najniższym poziomie korpusu

Jak wspomnieliśmy wcześniej, pliki text.xml są najcenniejszą częścią korpusu, gdyż to one zawierają teksty, oznakowane za pomocą wersji systemu XCES. Teksty te opisane są raczej zgrubnie: wyróżnione są ich poszczególne elementy strukturalne, takie jak sekcje, podsekcje, tytuły i poszczególne akapity, ale w tym miejscu segmentacja praktycznie się zatrzymuje. W obrębie akapitów, w zależności od postaci tekstów źródłowych, oznakowane mogą być ciągi wyróżnione np. kursywą, kapitalikami itp. Znakowanie tych ciągów jest także zgrubne, ponieważ dokonywane jest automatycznie: np. ciąg *de facto* znakowany jest jako <hi rend=”ital”>de facto</hi>, gdzie nazwa elementu powstała ze słowa *highlighted* ‘wyróżniony’, a nazwa atrybutu ze słowa *rendition*, co można tłumaczyć jako ‘forma wyróżnienia’. Funkcje wyróżnienia tekstu nie mogą być w przeważającej większości przypadków zidentyfikowane automatycznie, a więc pliki text.xml takich informacji nie zawierają. Dzięki rezygnacji ze szczegółowego oznakowania plików text.xml zyskujemy m.in. to, że tworzone są one w pełni automatycznie i że mogą natychmiast po utworzeniu funkcjonować jako pełnoprawne części całego korpusu.

Pliki text.xml ze swojej natury zajmują centralne miejsce w katalogach najniższego poziomu korpusu (zob. 2c). Poprzez system dowiązań odnoszą się do nich pliki sent.xml, a do tych z kolei pliki lex.xml, jak pokazano poniżej.

(4) text.xml <= sent.xml <= lex.xml ( <= disamb.xml )
|
header.xml

Plik sent.xml zawiera informacje o segmentacji zdaniowej wewnątrz każdego akapitu wyróżnionego w pliku text.xml. Plik lex.xml zawiera informacje o segmentacji dokonanej na potrzeby analizatora morfologicznego (zob. Woliński 2003) oraz wyniki jego działania. W obecnej formie korpusu IPI PAN plik ten zawiera również wyniki dezambiguacji, choć istnieje również możliwość, aby te informacje zachowywane były osobno, w pliku disamb.xml, powiązanym bezpośrednio z plikiem lex.xml. Ostatni z pokazanych w powyższym przykładzie plików to plik nagłówka, dowiązany do pliku text.xml.

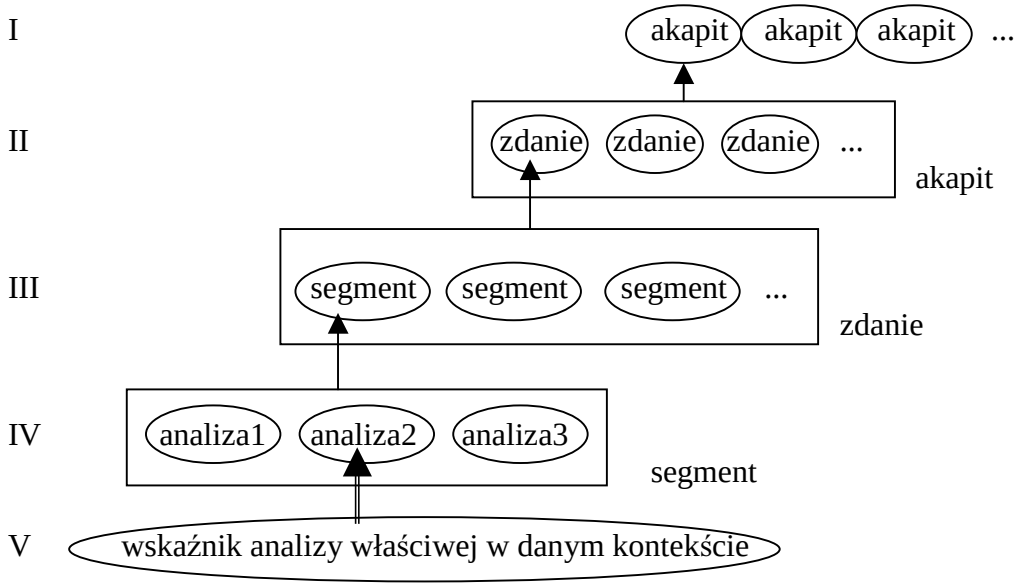
### 3.3. Struktura anotacji zewnętrznej

Anotacja zewnętrzna w korpusie IPI PAN może być także nazywana *anotacją warstwową*, ponieważ polega niejako na doklejaniu na plik text.xml kolejnych warstw opisu. Pierwsza warstwa anotacji zawarta jest w pliku text.xml. Następne warstwy mogą być ulokowane w różny sposób — w jednym pliku zewnętrznym, bądź ich większej ilości. Na rys. 5 pokazujemy ogólną strukturę takiego typu anotacji.

Kolejne poziomy anotacji połączone są jednostronnymi dowiązaniami, oznaczonymi za pomocą strzałek. Jak wspomnieliśmy, poziom I to poziom tekstu podstawowego. Minimalnym blokiem tekstu na tym poziomie jest akapit. Poziom II to poziom segmentacji zdaniowej, a poziom III — szeroko pojętego podziału na jednostki leksykalne (w przypadku korpusu IPI PAN segmentacja schodzi niekiedy poniżej poziomu słów — zob. Woliński 2003[11]). Poziom IV jest poziomem anotacji morfoskładniowej, a poziom V poziomem dezambiguacji.

[11] Dlatego też na rys. 5 używamy terminu *segment* — w znaczeniu prac Woliński (2003) i Przepiórkowski (2003), czyli jako tłumaczenie angielskiego *token*.

(5) Przykładowa struktura anotacji zewnętrznej




Zauważmy, że opis danych może mieć różnoraką naturę: na poziomach I-III podstawową rolą anotacji jest segmentacja tekstu — najpierw na jednostki strukturalne do poziomu akapitu, a następnie jednostki zdaniowe oraz leksykalne. Na poziomie IV do wyodrębnionych segmentów dołączony jest opis gramatyczny, a poziom V zawiera jedynie wskaźniki identyfikujące te spośród opisów proponowanych na poziomie IV, które akceptowane są w danym kontekście. Ten układ pozwala nam także na pokazanie pewnej istotnej właściwości anotacji zewnętrznej, a mianowicie przezroczystości jej poziomów: zauważmy, że z poziomu IV wskazujemy na elementy znajdujące się tak naprawdę na poziomie podstawowym — poziomy pośrednie są tu absolutnie transparentne. Inaczej możemy traktować wskazanie z poziomu V na poziom IV: może ono być zrealizowane jako wyodrębnienie (przynajmniej) jednego elementu poziomu IV, bez potrzeby sięgania gdziekolwiek dalej.

Rozdzielenie poziomów III i IV ma sens w językach, w których podział na jednostki podlegające opisowi pokrywa się z granicami słów (w takim wypadku lepiej jest wręcz połączyć poziomy II i III jako jeden poziom segmentacji). W języku polskim taki podział uzależniony jest od konkretnego analizatora morfologicznego. Analizator używany przez twórców korpusu IPI PAN (omówiony w pracy Woliński 2003) stosuje radykalną segmentację, rozdzielając opisywane jednostki nie tylko według spacji czy znaków interpunkcyjnych, ale oddzielając również np. tzw. klityki (*zrobili=ście*, *do=ń*). Dlatego też poziomy III i IV są w tym korpusie połączone.

Następna kwestia to kwestia łączenia poziomów IV i V. Z jednej strony, szczególnie w przypadku, gdy wypróbowywane są różne dezambiguatory (lub różne wersje tego samego dezambiguatora), takie rozdzielenie poziomów ma sens, gdyż ułatwia porównanie wyników lub formatów dezambiguacji.[12] Innym czynnikiem może być rodzaj samego dezambiguatora, w szczególności to, czy ma on jedynie wskazywać na wybrany opis morfosyntaktyczny, czy także przyznawać każdemu opisowi prawdopodobieństwo jego wystąpienia w danym kontekście. W tym drugim przypadku oczekiwalibyśmy, że dezambiguator zmodyfikuje jedynie pewne elementy poziomu IV, np. dodając odpowiednie atrybuty wyrażające stopień

[12] Robocze porównanie plików wynikowych różnych wersji tego samego dezambiguatora nie wymaga naturalnie konwersji na format XCES. Wyodrębnienie poziomu dezambiguacji (jako poziomu V) może natomiast służyć celom demonstracyjnym.

pewności, że dana forma jest w danym kontekście możliwa. W pierwszym przypadku, zgrabniejszym wyjściem wydaje się fizyczne rozdzielenie obydwu warstw.[13]

## 4. Anotacja zewnętrzna a efektywność tworzenia i wykorzystania korpusu

System CES od samego początku projektowany był z myślą o efektywności tworzenia i wykorzystania korpusów: wśród naczelnych zasad, którymi kierowali się jego twórcy, było zmniejszenie kosztów przetwarzania korpusu względem korpusów znakowanych w systemie TEI, rozbicie procesu tworzenia korpusu na etapy, oraz umożliwienie ponownego wykorzystania tak przygotowanego korpusu w przyszłości.

Modyfikacje, jakie wprowadzało CES względem TEI to drastyczne zawężenie możliwości użycia różnych sposobów znakowania dla tego samego tekstu, a także uproszczenie hierarchii typów danych oraz wprowadzenie nowych typów danych, dostosowanych do potrzeb językoznawstwa korpusowego (jest to opisane pokrótce w pracy Bański 2001, rozdz. 2.2-2.3, i szerzej w pracy głównych twórców CES, Ide i Véronis 1993).

Poniżej omawiamy kwestie podziału tworzenia korpusu na etapy (4.1) oraz jego rozszerzalności (4.2), a następnie sposobu, w jaki dzięki anotacji zewnętrznej można uniknąć problemu zazębiania się zakresów opisywanych danych (4.3).

### 4.1. Etapy tworzenia korpusu

CES proponuje trzy poziomy adekwatności znakowania korpusu (*conformance levels*, <http://www.cs.vassar.edu/CES/CES1-4.html>), jako propozycje standardu dla korpusów wykorzystywanych w inżynierii lingwistycznej. Charakteryzują się one rosnącą dokładnością znakowania korpusu przy proporcjonalnie rosnącym nakładzie pracy i środków pieniężnych. I tak, na przykład, aby dany tekst zaklasyfikowany został na poziomie pierwszym, nie może on zawierać obcych znaczników (rozumianych szeroko, także jako np. kody drukarskie w dokumentach Word Perfect itp.), podstawowe bloki tekstu muszą być konsekwentnie oznakowane, a nagłówek musi zawierać podstawowe informacje o sposobie przygotowania elektronicznej wersji danego tekstu. Na etapie drugim należy dodatkowo m.in. oznakować fragmenty zawarte w cudzysłowach i ustalić funkcję tekstu wyróżnionego (np. kursywą zwykle oznacza się emfazę, zwroty obcojęzyczne, cytaty, itd.), podczas gdy poziom trzeci to poziom największej dokładności i jednorodności znakowania.

Nowo skompilowane teksty korpusu IPI PAN z założenia spełniają kryteria klasyfikacji na poziomie pierwszym, a więc już w momencie dodania ich do korpusu (tj. ‘wpięcia’ ich w strukturę korpusu za pomocą systemu opartego na całostkach XML-a), zyskujemy w nich podstawowy materiał do standardowych badań lingwistycznych. Tak więc, nawet na wczesnych etapach tworzenia korpusu, gdy w tekstach nie wydzielone są jeszcze segmenty zdaniowe i gdy nie istnieje jeszcze anotacja lingwistyczna tychże tekstów, architektura korpusu IPI PAN zapewnia możliwość jego skutecznego wykorzystania. Dzieje się tak, ponieważ dany zbiór tekstów tworzy całość już po wstępnej, automatycznej obróbce, gdy składa się wyłącznie z plików text.xml (oraz ich nagłówków). Już wtedy można go wykorzystywać w celu uzyskania np. prostych konkordancji czy list frekwencyjnych. To samo odnosi się do rozszerzania korpusu w przyszłości, o nowe teksty i ich dowolnego rodzaju anotacje. Dzięki informacjom zawartym w nagłówkach tekstów istnieje też mechanizm pozwalający włączać lub wykluczać daną porcję tekstów, w zależności od tego,

[13] Jest to tylko jeden z czynników rozważanych przy wyborze takiej implementacji. Innym jest np. rozmiar korpusu powiększonego o dodatkową warstwę plików disamb.xml — oznaczać to może dodanie setek plików o średnich rozmiarach od kilku do kilkunastu kilobajtów.

czy do danego zadania potrzeba tekstu anotowanego szczegółowo, czy też chodzi o sam czysty tekst.

Samo to, że nie trzeba każdego tekstu posegmentować i opisać gramatycznie przed włączeniem go do korpusu stwarza lepsze możliwości zarządzania środkami finansowymi i pracą anotatorów. Jak wspomnieliśmy, pliki text.xml tworzone są automatycznie i praktycznie bez nadzoru, a więc włączenie ich do korpusu nie jest spowolnione przez 'czynnik ludzki'. Dopiero po zainstalowaniu plików text.xml w odpowiednich katalogach można automatycznie wyprodukować pliki sent.xml, zawierające segmentację zdaniową oraz odniesienia do elementów <hi> (wyróżnionych). Pliki w takiej formie prezentowane są anotatorowi-językoznawcy, który i) sprawdza poprawność podziału na zdania i ii) klasyfikuje funkcję elementów wyróżnionych (jako emfaza, zwrot obcy itd.).

Następnie każdy tekst jest automatycznie przetwarzany na postać wejściową do analizatora morfologicznego i dezambiguatora, który na wyjściu tworzy plik lex.xml.[14]

### 4.2. Rozszerzalność korpusu

Jak podkreślają Ide i Brew (2000), rozszerzalność korpusu, rozumiana zarówno jako możliwość ponownego użycia jego komponentów jak i możliwość jego rozbudowy (włączając w to także łatwość dostosowania go do jeszcze nie przewidzianych zadań), nie jest już kwestią kaprysu jego twórców, ale jednym z podstawowych kryteriów jego oceny oraz opłacalności. Wprowadzenie anotacji zewnętrznej znacznie upraszcza proces udoskonalania i rozbudowy korpusu, gdyż stwarza możliwość ulokowania kolejnych etapów uściślania opisu tekstu na osobnych poziomach anotacji, oznakowanych w osobnych plikach.

Istnieje oczywisty konflikt pomiędzy stopniem specjalizacji anotacji korpusu a zakresem jego zastosowań. Przyjęcie anotacji zewnętrznej pozwala na uniknięcie tego konfliktu pod jednym, trywialnym, warunkiem: że tekst główny będzie anotowany jak najbardziej ogólnie — tak, aby nie wyeliminować żadnego rodzaju zastosowań. Cała reszta to kwestia dobudowania zewnętrznych warstw anotacji odpowiednich dla zadanych wymagań.

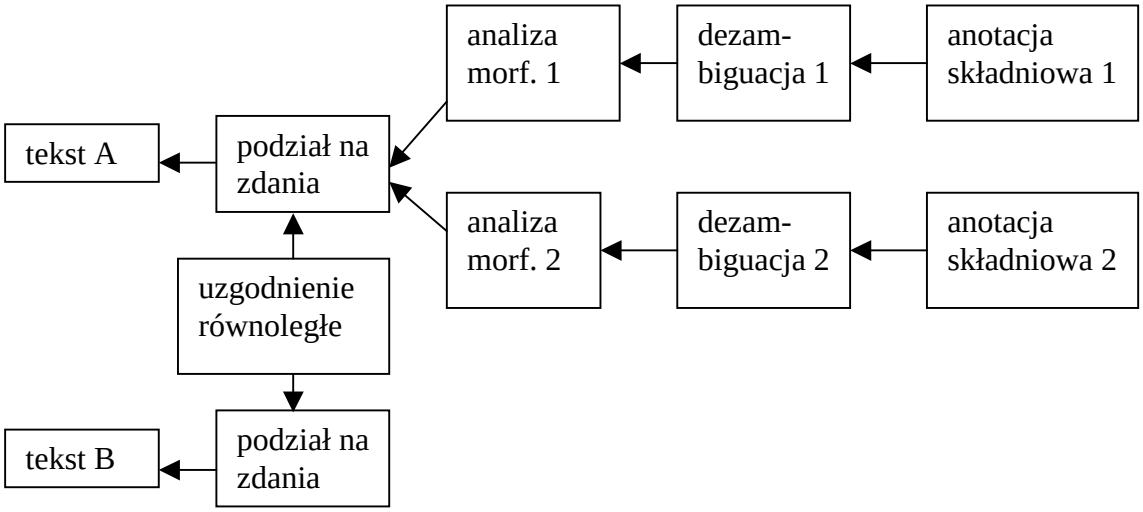
Ponieważ dodawanie nowych warstw anotacji zewnętrznej oznacza modyfikację lub (częściej) dodawanie nowych plików, odwołujących się pośrednio lub bezpośrednio do pliku text.xml, plik ten powinien być zabezpieczony przed zapisem — w ten sposób tekst główny jest chroniony przed przypadkowym uszkodzeniem lub zniszczeniem, podczas gdy fakt dodania kolejnej warstwy anotacji zaznaczany jest w wydzielonym z tekstu głównego nagłówku (header.xml). Dołączanie nowych warstw anotacji nie wymaga żadnych modyfikacji w warstwach już istniejących — odpowiedzialność za właściwe 'wpięcie się' w strukturę korpusu spoczywa wyłącznie na nowo dołączanym pliku, np. z anotacją składniową — musi on 'dokleić się' za pomocą dowiązań do anotacji na poziomie ujednoznaczniania (czyli na poziomie V z rys. 5 — pliku lex.xml lub disamb.xml, gdy taki istnieje), por. rys. 6.[15]

Schemat pokazany na rys. 6 ilustruje także możliwość 'recyklingu' materiałów składowych korpusu. Jest to znacznie uproszczone przez zastosowanie anotacji zewnętrznej — w ten sposób tekst główny wraz z częścią plików dodatkowych, np. plikiem segmentacji zdaniowej, może być użyty bezpośrednio w nowym projekcie, na przykład do utworzenia korpusu wielojęzycznego: równoległego lub porównawczego.

[14] Pomijamy pewne szczegóły tego procesu. Warto też zauważyć, że jeśli narzędzie używane do segmentacji zdaniowej działa z dużą bezbłędnością, interwencja anotatora może być przesunięta na etap późniejszy. Dodajmy jeszcze, że każdy zabieg wewnątrz katalogu zawierającego tekst i pliki dodatkowe musi być odzierciedlony w nagłówku, przeważnie w sposób automatyczny.

[15] Anotacja zewnętrzna nie musi jednak składać się z wielu warstw: możliwy teoretycznie jest układ, w którym każdy plik dodatkowy odnosi się bezpośrednio do pliku centralnego. Możliwy też jest układ mieszany, jak pokazano na rys. 6.

(6) Możliwe rozszerzenie konstrukcji korpusu




Dalszym ułatwieniem, powiązanym z kwestią rozszerzalności, jest możliwość zainstalowania np. kilku plików z różnymi anotacjami składniowymi odnoszącymi się do tego samego pliku z anotacją morfosyntaktyczną — w ten sposób można np. porównywać różne teorie składniowe, także poprzez wizualne nakładanie na siebie konstrukcji przez nie przewidywanych.

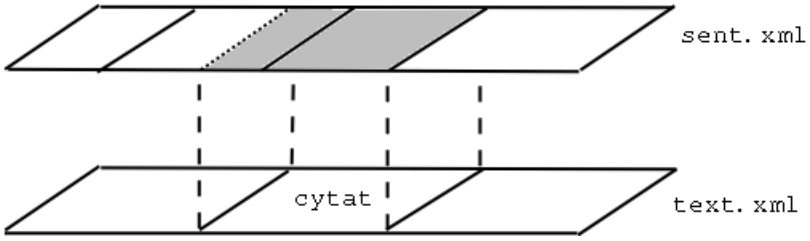
### 4.3. Zazębiające się zakresy danych

Anotacja zewnętrzna ma jeszcze jedną istotną zaletę: pozwala uniknąć problemu zazębiania się zakresów danych. Ponieważ dokument XML ma strukturę drzewiastą, nie jest możliwe oznakowanie zdania w przykładzie (7a) za pomocą elementów <s> (=zdanie) oraz <q> (=cytat).

(7) a. Powiedziała, że on „ma tego dość. I nie chce tu przychodzić”.
b. <s>Powiedziała, że on <q>ma tego dość.</s><s>I nie chce tu przychodzić</q>.</s>

Oznakowanie pokazane w (7b) nie jest poprawnym przykładem XML-a, gdzie elementy nie mogą się zazębiać. Tak więc, aby oznaczyć w jednym dokumencie i podział na zdania, i cytaty, trzeba używać technicznych trików. Jeśli natomiast posługujemy się anotacją zewnętrzna, wystarczy takie zazębiające się elementy umieścić w różnych plikach. W korpusie IPI PAN, elementy <q> umieszczone są w plikach text.xml, a <s> — w sent.xml:[16]

(8)




[16] Używamy tu elementów <s> i <q> jako wygodnego przykładu radzenia sobie z tzw. paradoksami nawiasowania (ang. *bracketing paradoxes*), występującymi czasem przy pełnym opisie gramatycznym, gdy zazębia się np nawiasowanie semantyczne i morfologiczne. W rzeczywistości, z przyczyn technicznych

W ten sposób oba zakresy danych nakładają się już tylko przy wizualizacji tekstów, co nie stwarza żadnych komplikacji.

## 5. Podsumowanie

Omówiliśmy tu podstawowe zasady architektury korpusu IPI PAN oraz ich wpływ na maksymalizację korzyści, jakie tego typu korpus może przynieść.

Należy podkreślić, iż korpus ten będzie, obok American National Corpus (<http://americannationalcorpus.org/>), jednym z pierwszych korpusów na świecie wykorzystujących najnowsze wytwory postępu technologicznego i naukowego w dziedzinie architektury danych tekstowych oraz ich przetwarzania w oparciu o standardy XML.

## Bibliografia

Bański, Piotr (2001). The proposed encoding scheme for the IPI PAN corpus. Raport techniczny, Warszawa: Instytut Podstaw Informatyki PAN.

Bray, Tim, Jean Paoli, C. M. Sperberg-McQueen, Eve Maler (2000). *Extensible Markup Language (XML) 1.0* (wyd. drugie) Rekomendacja Konsorcjum W3C <http://www.w3.org/>.

Ide, Nancy i Chris Brew (2000). Requirements, Tools, and Architectures for Annotated Corpora. W materiałach konferencji *Data Architectures and Software Support for Large Corpora*. Paryż: European Language Resources Association, 1-5.

Ide, Nancy i Jean Véronis (1993). Background and Context for the Development of a Corpus Linguistic Standard. Raport projektu EAGLES, dostępny pod adresem <http://www.cs.vassar.edu/CES/CES3.ps.gz>.

McEnery, Tony i Andrew Wilson (2001). *Corpus Linguistics*. Edynburg: Edinburgh University Press.

Przepiórkowski, Adam (2003). Dehomonimizacja w korpusie IPI PAN. Niniejszy tom.

Przepiórkowski, Adam, Piotr Bański, Łukasz Dębowski, Elżbieta Hajnicz i Marcin Woliński, 2003, Konstrukcja korpusu IPI PAN. Niniejszy tom.

*Słownik naukowo-techniczny angielsko-polski*, 1990, red. Maria Skrzyńska, Sergiusz Czerni, Teresa Jaworska i Ewa Romkowska. Warszawa: Wydawnictwa Naukowo Techniczne.

Woliński, Marcin (2003). System znaczników morfosyntaktycznych w korpusie IPI PAN. Niniejszy tom.

elementy <q> nie są zwykle umieszczane w tekście podstawowym w pierwszej fazie obróbki, chyba że format tekstu źródłowego pozwala na ich automatyczną identyfikację.