

Przetwarzanie wstępne.

Pierwszym krokiem przy tworzeniu elektronicznej wersji słownika było zeskanowanie wszystkich stron książkowej jego wersji. Otrzymane w ten sposób obrazy przetworzono programem OCR, w wyniku czego utworzony został plik w formacie Microsoft Word. Następnym krokiem było wyodrębnienie z tego pliku wszystkich haseł słownika wraz z opisami i zapisanie ich w postaci tekstowej (UTF-8).

W tym celu stworzony został program, który na samym początku wydzielił główną część słownika zawierającą opisywane hasła od wstępu, strony tytułowej i stron redakcyjnych. Na tym etapie przetwarzania program wykorzystał układ tekstu na stronach słownika, który w części zawierającej hasła (i tylko w niej) jest układem dwukolumnowym.

Identyfikacja haseł nastręczyła więcej problemów. W wersji książkowej hasła oraz ich opisy umieszczone są w jednym paragrafie, przy czym definiowane hasło, jak również podhasła, wyróżnione są pogrubioną czcionką. Niestety, nie najlepsza jakość skanów spowodowała, że program OCR często nie rozpoznawał czy użyta została pogrubiona, czy też zwyczajna czcionka. Spowodowało to w niektórych przypadkach sklejanie kilku haseł w jedno lub podzielenie tekstu dotyczącego jednego hasła. Pomocną informacją, która w wielu przypadkach pozwoliła na prawidłową identyfikację haseł, była informacja dotycząca głębokości wcięcia wiersza tekstu. Każde rozpoznane hasło wraz z podhasłami, ich definicjami i przykładami użycia, zostało zapisane w jednej linii tekstu w pliku tekstowym przy użyciu kodowania UTF-8. Dodatkowo, hasła zostały odseparowane od siebie pustymi liniami.

Tak utworzony plik wymagał ręcznego poprawienia z uwagi na wspomnianą już wyżej złą jakość skanów. Ponadto sam program OCR, przystosowany do współczesnego języka polskiego, zastępował niekiedy nierozpoznane sekwencje znaków innymi, występującymi w jego słowniku. Najczęstszymi poprawkami, które musiały być wykonane ręcznie, było wstawianie lub usuwanie znaków diakrytycznych, wstawianie znaków '«' i '»', odpowiednio na początku i końcu definicji, zgodnie z konwencją przyjętą w słowniku oraz usuwanie znaków nieistniejących w tekście a powstałych w wyniku interpretacji przez program OCR różnego rodzaju zabrudzeń i skaz papieru jako ciągu znaków.

Konwersja na format LMF.

Konwersja wersji tekstowej na wersję w formacie LMF przeprowadzana jest w kilku etapach. W pierwszym etapie następuje klasyfikacja haseł do odrębnych klas. Opisy haseł należących do jednej klasy mają podobną strukturę. W drugim kroku następuje właściwa konwersja zgodnie z określonymi dla danej klasy haseł regułami. Po przetworzeniu tekstu całego słownika następuje faza trzecia, w której ustalane są identyfikatory tych haseł, do których istnieje odwołanie w tekście innych haseł. Usuwane są wtedy redundantne wpisy, jak również tworzona jest lista haseł niezidentyfikowanych, do których odwołuje się w tekście, a które nie znalazły się w słowniku. Kolejną fazą, która jeszcze wymaga zaimplementowania, jest uzupełnienie każdego wpisu w słowniku o brakujące informacje morfologiczne. Wersja książkowa na ogół nie posiada takich informacji. Nie jest określona część mowy opisywanego hasła, a odmienione formy są przedstawione tylko w sposób skrótowy, poprzez podanie samych końcówek i to tylko w tych przypadkach, w których odmiana jest nietypowa lub różni się od tej spotykanej we współczesnej polszczyźnie. Ostatnią fazą jest ręczna weryfikacja wygenerowanych struktur LMF.

Należy zaznaczyć, że w książkowej wersji słownika brakuje jasno określonej konwencji opisu haseł. Co więcej, przyjęte reguły nie są ściśle przestrzegane. W wielu przypadkach z pozoru podobnie wyglądające zapisy mają różne znaczenia, co nieuchronnie prowadzi do błędnej translacji. Błędy mogą się również pojawić przy automatycznym uzupełnianiu słownika o brakujące informacje morfologiczne. Dlatego ostatnia faza, jaką jest ręczna weryfikacja, jest niezbędna. W tym celu zaplanowana jest w niedalekiej przyszłości implementacja edytora struktur LMF.

Struktura słownika.

Konstruowany słownik w formacie LMF jest plikiem XML o następującej strukturze:

```
<?xml version="1.0" encoding="UTF-8"?>
<LexicalResource dtdVersion="16">
  <GlobalInformation>
    <feat att="languageCoding" val="ISO 639-6" />
  </GlobalInformation>
  <Lexicon>
    <feat att="language" val="polh" />
    <LexicalEntry id="lex.1">
      ...
    </LexicalEntry>
    ...
    <LexicalEntry id="lex.n">
      ...
    </LexicalEntry>
  </Lexicon>
</LexicalResource>
```

Główną częścią tej struktury jest leksykon (<Lexycon>) stanowiący zbiór haseł leksykalnych (<LexicalEntry>), z których każde hasło reprezentuje jeden leksem języka historycznego (polh). Każdy element <LexicalEntry> posiada atrybut o nazwie id określający unikalny identyfikator hasła w leksykonie. Ogólna postać wartości tego atrybutu jest następująca: lex.n lub lex.n:m, gdzie n i m są dodatkowymi liczbami całkowitymi i oznaczają numer hasła głównego (n) oraz numer podhasła (m). Każde hasło reprezentowane przez element <LexicalEntry> opatrzone jest komentarzem:

```
<!--INDEX nnn -->,
```

gdzie liczba nnn oznacza numer linii w wejściowym pliku tekstowym, w której dane hasło zostało umieszczone. Dodatkowo, w komentarzu może się znajdować słowo VERIFY, ale tylko w tych przypadkach, w których konieczna jest weryfikacja poprawności translacji przez lingwistę.

Z każdym hasłem związana jest jedna forma podstawowa (<Lemma>), oraz zbiór form wyrazowych (<WordForm>). Z każdą formą związana jest jej reprezentacja w języku pisanym (<FormRepresentation>). W szczególności, jeśli dane słowo posiada kilka wariantów ortograficznych, będą one przedstawione w osobnych instancjach klasy <FormRepresentation>. Leksem może posiadać wiele znaczeń. Każde z nich jest reprezentowane przez klasę <Sense>. Znaczenie leksemu może być określone za pomocą definicji (<Definition>) (wraz z uzupełnieniem w polu <Statement>), lub

poprzez podanie przykładowych kontekstów użycia słowa (klasa <Context>). W każdym z tych przypadków do opisu znaczenia leksemu służy klasa <TextRepresentation>.

Użycie powyższych oraz innych klas języka LMF zostanie zaprezentowane w przykładach.

Reguły translacji.

Około 82% wszystkich haseł w przetwarzanym słowniku ma następujący wzorzec:

```
{słowo} «definicja»: cytowania.
```

lub

```
{słowo} 1. «definicja-1»: cytowania-1;  
        2. «definicja-2»: cytowania-2;  
        ...  
        n. «definicja-n»: cytowania-n.
```

Zaraz po zdefiniowanym słowie ujętym w nawiasy klamrowe następuje jeden lub więcej ponumerowanych bloków. Każdy z tych bloków zaczyna się definicją otoczoną nawiasami kątowymi '«' i '»', po której występuje dwukropek a następnie przykłady użycia zdefiniowanego wyrazu rozdzielone średnikami. Prawie każdy przykład zawiera na samym końcu informację na temat źródła pochodzenia.

Przykład 1.

```
{abdankować} 1. «odprawić, rozpuścić (o wojaku, służbie)»: Wojsk nie chciał  
z siebie abdankować L XVII; 2. «rezygnować»: Chciwość hamować i rzeczom  
cudzym abdankować L XVII.
```

Hasło to ma następującą postać LMF:

```
<LexicalEntry id="lex.1">  
  <!-- INDEX 1 -->  
  <Lemma>  
    <FormRepresentation>  
      <feat att="writtenForm" val="abdankować" />  
      <feat att="language" val="polh" />  
      <feat att="sourceID" val="srpsdp:L XVII" />  
    </FormRepresentation>  
  </Lemma>  
  <Sense>  
    <Definition>  
      <TextRepresentation>  
        <feat att="writtenForm"  
              val="odprawić, rozpuścić (o wojaku, służbie)" />  
        <feat att="language" val="pol" />  
        <feat att="sourceID" val="srpsdp" />  
      </TextRepresentation>  
    </Definition>  
  <Context>  
    <TextRepresentation>
```

```

        <feat att="writtenForm"
            val="Wojsk nie chciał z siebie abdikować" />
        <feat att="language" val="polh" />
        <feat att="sourceID" val="srpsdp:L XVII" />
    </TextRepresentation>
</Context>
</Sense>
<Sense>
    <Definition>
        <TextRepresentation>
            <feat att="writtenForm" val="rezygnować" />
            <feat att="language" val="pol" />
            <feat att="sourceID" val="srpsdp" />
        </TextRepresentation>
    </Definition>
    <Context>
        <TextRepresentation>
            <feat att="writtenForm"
                val="Chciwość hamować i rzeczom cudzym abdikować" />
            <feat att="language" val="polh" />
            <feat att="sourceID" val="srpsdp:L XVII" />
        </TextRepresentation>
    </Context>
</Sense>
</LexicalEntry>

```

Nie wszystkie hasła pasują do powyższego wzorca. Przykładem mogą być hasła wielowyrazowe.

Przykład 2.

{zableszczyć oczy} «przewrócić oczyma, spojrzeć bokiem, błysnąć, łypnąć oczyma»: Zableszczysz na strony oczy RPolik.

Hasło to ma następującą postać LMF:

```

<LexicalEntry id="lex.20112">
  <!-- INDEX 16833 VERIFY -->
  <Lemma>
    <FormRepresentation>
      <feat att="writtenForm" val="zableszczyć oczy" />
      <feat att="language" val="polh" />
      <feat att="sourceID" val="srpsdp:RPolik" />
    </FormRepresentation>
  </Lemma>
  <ListOfComponents>
    <Component target="lex.xxxx" />
    <Component target="lex.yyyy" />
  </ListOfComponents>
  <Sense>
    <Definition>
      <TextRepresentation>
        <feat att="writtenForm" val="przewrócić oczyma, spojrzeć bokiem,
            błysnąć, łypnąć oczyma" />
      </TextRepresentation>
    </Definition>
  </Sense>
</LexicalEntry>

```

```

        <feat att="language" val="pol" />
        <feat att="sourceID" val="srpsdp" />
    </TextRepresentation>
</Definition>
<Context>
    <TextRepresentation>
        <feat att="writtenForm" val="Zableszczysz na strony oczy" />
        <feat att="language" val="polh" />
        <feat att="sourceID" val="srpsdp:RPolik" />
    </TextRepresentation>
</Context>
</Sense>
</LexicalEntry>

```

Identyfikatory `lex.xxxx` i `lex.yyyy` są identyfikatorami haseł „zableszczyć” oraz „oko”, które powinny znajdować się w leksykonie. Niestety w wielu przypadkach elementy składowe haseł wielowyrazowych nie zostały w słowniku uwzględnione. Tego rodzaju luki powinny być w przyszłości uzupełnione. Trudność identyfikacji haseł składowych polega na tym, że mogą one występować w haśle wielowyrazowym w postaci odmienionej. Jeśli składowa `component` hasła złożonego nie występuje w słowniku dokładnie w takiej formie, w jakiej występuje w tym haśle, program automatycznie dołącza do słownika następujący element:

```

<LexicalEntry id="identyfikator składowej 'component'">
    <!-- VERIFY -->
    <Lemma>
        <FormRepresentation>
            <feat att="writtenForm" val="component" />
            <feat att="language" val="polh" />
            <feat att="sourceID" val="srpsdp" />
        </FormRepresentation>
    </Lemma>
</LexicalEntry>

```

Taki wpis musi być następnie zweryfikowany.

Pokażną grupę haseł wielowyrazowych stanowią czasowniki zwrotne, czyli takie, które zawierają zaimek zwrotny „się”. Zaimek ten jest nieodłączną częścią czasownika i dlatego dla takich haseł nie jest tworzona struktura `<ListOfComponents>`.

Przykład 3.

{ekskuzować się} «usprawiedliwiać, wymawiać się»: Ekskuzował się i wywodził PSk.

```

<LexicalEntry id="lex.2987">
    <!-- INDEX 2523 -->
    <Lemma>
        <FormRepresentation>
            <feat att="writtenForm" val="ekskuzować się" />
            <feat att="language" val="polh" />
            <feat att="sourceID" val="srpsdp:PSk" />
        </FormRepresentation>
    </Lemma>
</LexicalEntry>

```

```

    </FormRepresentation>
  </Lemma>
  <Sense>
    <Definition>
      <TextRepresentation>
        <feat att="writtenForm" val="usprawiedliwiać, wymawiać się" />
        <feat att="language" val="pol" />
        <feat att="sourceID" val="srpsdp" />
      </TextRepresentation>
    </Definition>
    <Context>
      <TextRepresentation>
        <feat att="writtenForm" val="Ekskuzował się i wywodził" />
        <feat att="language" val="polh" />
        <feat att="sourceID" val="srpsdp:PSk" />
      </TextRepresentation>
    </Context>
  </Sense>
</LexicalEntry>

```

Czasowniki zwrotne często nie mają swojej definicji lecz dzielą ją z formą niezwrotną.

Przykład 4.

{chycić, ~ się} «chwycić (się)»: Chyciwszy się rękoma jej gardła, wywiedli ją BkZ; Chyciłam się go BLEop; HTroj.

```

<LexicalEntry id="lex.1344">
  <!-- INDEX 1268 VERIFY -->
  <Lemma>
    <FormRepresentation>
      <feat att="writtenForm" val="chycić" />
      <feat att="language" val="polh" />
      <feat att="sourceID" val="srpsdp" />
    </FormRepresentation>
  </Lemma>
  ...
</LexicalEntry>

<LexicalEntry id="lex.1344:1">
  <!-- INDEX 1268 VERIFY -->
  <Lemma>
    <FormRepresentation>
      <feat att="writtenForm" val="chycić się" />
      <feat att="language" val="polh" />
      <feat att="sourceID" val="srpsdp" />
    </FormRepresentation>
  </Lemma>
  <RelatedForm target="lex.1344">
    <feat att="writtenForm" val="chycić" />
    <feat att="language" val="polh" />
    <feat att="sourceID" val="srpsdp" />
  </RelatedForm>

```

```

    </RelatedForm>
</LexicalEntry>

```

Przykład 5.

{gibać (się)} «giać (się) kręcić, zginać»: Kręcac a. gibając exterminans
SStp; Gibam się, przegibam się GKn.

```

<LexicalEntry id="lex.3032">
  <!-- INDEX 1268 VERIFY -->
  <Lemma>
    <FormRepresentation>
      <feat att="writtenForm" val="gibać" />
      <feat att="language" val="polh" />
      <feat att="sourceID" val="srpsdp" />
    </FormRepresentation>
  </Lemma>
  ...
</LexicalEntry>

<LexicalEntry id="lex.3032:1">
  <!-- INDEX 1268 VERIFY -->
  <Lemma>
    <FormRepresentation>
      <feat att="writtenForm" val="gibać się" />
      <feat att="language" val="polh" />
      <feat att="sourceID" val="srpsdp" />
    </FormRepresentation>
  </Lemma>
  <RelatedForm target="lex.3032">
    <feat att="writtenForm" val="gibać" />
    <feat att="language" val="polh" />
    <feat att="sourceID" val="srpsdp" />
  </RelatedForm>
</LexicalEntry>

```

W bardziej złożonych hasłach definiowane są także jego podhasła.

Przykład 6.

{plac} «miejsce»: Potym dobra cielesne plac mają za tymi JKoch; {~ dać}
«wstąpić»: SPetr; {~ otrzymać} «zwyctężyć, otrzymać pierwszeństwo»: Lepsza
była rada pierwsza, ale pośledniejsza plac otrzymała MBiel; JKoch.

W tym przykładzie występują dwa podhasła tworzące związki frazeologiczne z wyrazem "plac". Zostały one umieszczone w osobnych wpisach w tworzonym słowniku.

```

<LexicalEntry id="lex.10144">
  <!-- INDEX 9315 VERIFY -->
  <Lemma>
    <FormRepresentation>
      <feat att="writtenForm" val="plac" />

```

```

    ...
    </FormRepresentation>
  </Lemma>
  ...
</LexicalEntry>

<LexicalEntry id="lex.10144:1">
  <!-- INDEX 9315 VERIFY -->
  <Lemma>
    <FormRepresentation>
      <feat att="writtenForm" val="plac dać" />
      ...
    </FormRepresentation>
  </Lemma>
  <ListOfComponents>
    ...
  </ListOfComponents>
  ....
</LexicalEntry>

<LexicalEntry id="lex.10144:2">
  <!-- INDEX 9315 VERIFY -->
  <Lemma>
    <FormRepresentation>
      <feat att="writtenForm" val="plac otrzymać" />
      ...
    </FormRepresentation>
  </Lemma>
  <ListOfComponents>
    ...
  </ListOfComponents>
  ....
</LexicalEntry>

```

Przykład 7.

{błaźnić} «mamić, oszukiwać, otumaniać, ogłupiać»: Nie dajcie sie błaźnić żonom swoim JWuj; Ich umiejętność błaźnię Bar; {~ z kogo} «drwić, kpić z kogo»: Naśmiewam się, błaźnię z kogo. Błaźnisz ze mnie SMącz.

Hasło „błaźnić z kogo” jest przykładem rekcji. Rekcja reprezentowana jest przez element <SyntacticBehaviour>.

```

<LexicalEntry id="lex.704">
  <!-- INDEX 629 VERIFY -->
  <Lemma>
    <FormRepresentation>
      <feat att="writtenForm" val="błaźnić" />
      ...
    </FormRepresentation>
  </Lemma>
  <SyntacticBehaviour senses="sense.179">

```



```

    <feat att="writtenForm" val="błaźnić z kogo" />
    <feat att="language" val="polh" />
    <feat att="sourceID" val="srpsdp:SMącz" />
</SyntacticBehaviour>
<Sense>
  <Definition>
    <TextRepresentation>
      <feat att="writtenForm"
        val="mamić, oszukiwać, otumaniać, ogłupiać" />
      <feat att="language" val="pol" />
      <feat att="sourceID" val="srpsdp" />
    </TextRepresentation>
  </Definition>
  <Context>
    <TextRepresentation>
      <feat att="writtenForm" val="Nie dajcie sie błaźnić żonom swoim" />
      <feat att="language" val="polh" />
      <feat att="sourceID" val="srpsdp:JWuj" />
    </TextRepresentation>
  </Context>
  <Context>
    <TextRepresentation>
      <feat att="writtenForm" val="Ich umiejętność błaźnić" />
      <feat att="language" val="polh" />
      <feat att="sourceID" val="srpsdp:Bar" />
    </TextRepresentation>
  </Context>
</Sense>
<Sense id="sense.179">
  <Definition>
    <TextRepresentation>
      <feat att="writtenForm" val="drwić, kpić z kogo" />
      <feat att="language" val="pol" />
      <feat att="sourceID" val="srpsdp" />
    </TextRepresentation>
  </Definition>
  <Context>
    <TextRepresentation>
      <feat att="writtenForm"
        val="Naśmiewam się, błaźnię z kogo. Błaźnisz ze mnie" />
      <feat att="language" val="polh" />
      <feat att="sourceID" val="srpsdp:SMącz" />
    </TextRepresentation>
  </Context>
</Sense>
</LexicalEntry>

```

Niektóre hasła zawierają listę wariantywnych jego form, co ilustrują następujące trzy przykłady.

Przykład 8.

{obiecadło, obiecado, abiecadło} «abecadło, alfabet»: A jesliby w słowie litery z obiecadła niedostawało KDzik; Kiedy przyszedł k temu słowu w

obiecadle RPrz; Umiej obiecado moje SStp; I czytając zaś one obiecadła PKoch; Porządkiem polskim abiecadła BBud; MRej; JKoch; FrMiesz.

Wszystkie warianty są wymienione w elemencie <Lemma> struktury LMF.

```
<LexicalEntry id="lex.7962">
  <-- INDEX 7503 VERIFY -->
  <Lemma>
    <FormRepresentation>
      <feat att="writtenForm" val="obiecadło" />
      <feat att="language" val="polh" />
      <feat att="sourceID" val="srpsdp:KDzik:RPrz:PKoch" />
    </FormRepresentation>
    <FormRepresentation>
      <feat att="writtenForm" val="obiecado" />
      <feat att="language" val="polh" />
      <feat att="sourceID" val="srpsdp:SStp" />
    </FormRepresentation>
    <FormRepresentation>
      <feat att="writtenForm" val="abiecadło" />
      <feat att="language" val="polh" />
      <feat att="sourceID" val="srpsdp:BBud:MRej:JKoch:FrMiesz" />
    </FormRepresentation>
  </Lemma>
  ...
</LexicalEntry>
```

Przykład 9.

{zachodźca, -e || zachojsca, -e} «wyręczyciel, zastępca»: Już masz pewnego zachojscę za występki swoje MRej; A on by zachojsce dostawić nie mógł, od kogo by to miał APrM; Mamy zachojscę Pana Jezusa GilP.

W tym przykładzie zdefiniowane są dwie wariantywne formy „zachodźca” i „zachojsca”.

Przykład 10.

{chlebojedźca} «famiłiant, domownik, podopieczny»: Czeladkaśmy boska i chlebojedźcy domowi jego PSk; Abramow chlebojedźca pobrał koły RWar; {chlebojedziec}: Na szlachcicu, na swem chlebojedźcu SStp.

Podhasło „chlebojedziec” nie ma własnej definicji i dlatego zostaje uznane jako wariantywna forma hasła „chlebojedźca”.

```
<LexicalEntry id="lex.1133">
  <-- INDEX 1082 VERIFY -->
  <Lemma>
    <FormRepresentation>
      <feat att="writtenForm" val="chlebojedźca" />
      ...
    </FormRepresentation>
    <FormRepresentation>
```

```

        <feat att="writtenForm" val="chlebojedziec" />
        ...
    </FormRepresentation>
    ...
    <Lemma>
</LexicalEntry>

```

Czasami wariantywna postać hasła określona jest implicite poprzez ujęcie części tekstu w nawiasy.

Przykład 11.

```

{is(t)ność} 1. «istota, natura»: Jisność, bytnoś(ć) essencia SStp; Byłem na
początku zawsze u boga ojca... a byłem rowien ojcu memu w tej isności
wiekuistej RPrz; Essencyja bóstwa, to jest istność MRej; JKoch; 2.
«kapitał»: Quia vere existencie al. istności eadem peccunia est SStp.

```

W tym hasle zdefiniowane są dwie równoważne formy „isność” i „istność”.

Zdarza się, że wariantywne postaci definiowane są w słowniku jako osobne hasła.

Przykład 12.

```

{chlebojejca, -e} «podopieczny, najemnik»: Jesteśmy tu gośćmi... i
chlebojejcami jego JŻeb; SEBr.

```

Hasło „chlebojejca” ma to samo znaczenie, co „chłobojedźca” i „chlebojedziec”. Wydaje się zatem, że powinno być traktowane jako równoważna forma tych haseł. O tym jednak musi zdecydować lingwista na etapie ręcznego poprawiania słownika.

Jak już wcześniej zostało to wspomniane, książkowa wersja słownika praktycznie nie zawiera żadnych informacji morfologicznych. Jeśli już takie występują, to są przedstawione bardzo skrótowo tak jak w powyższym przykładzie. Należy się domyślać, że chodzi w tym przypadku o mianownik liczby mnogiej hasła „chłobojejca”, tzn. o formę „chlebojeje”. W takich przypadkach program dołącza do elementu <LexicalEntry> strukturę <WordForm>:

```

<WordForm>
  <FormRepresentation>
    <feat att="writtenForm" val="chlebojejca-e" />
    <feat att="language" val="polh" />
    <feat att="sourceID" val="srpsdp" />
  </FormRepresentation>
</WordForm>

```

Tę strukturę należy następnie poprawić i uzupełnić, co w części może być wykonane w zautomatyzowany sposób.

Podobnie program traktuje wpisy zawierające ciąg znaków „||”.

Przykład 13.

{zakon, -a || -u} 1. «prawo, ustawa»: Przeto zakon da zostającym na drodze PFl; Zakonów mych ostrzegajcie BkZ; W synowskim swej matki zostaje zakonie WPot; 2. «religia»: A gdyż wam by rzekli synowie waszy: kaki jest to zakon BkZ; Apostata, ten, co z zakonu do zakonu przystępuje ERot; 3. «stan»: Jest zakon wdowi, małżeński i paniński MRej; JKoch.

W tym przypadku dołączone zostaną dwie struktury <WordForm> odpowiadające formom „zakona” (zakon-a) i „zakonu” (zakon-u).

Osobną klasą haseł są hasła, które składają się z dwóch członów rozdzielonych znakiem średnika lub dwukropkiem. W tych przypadkach, zarówno definicje jak i cytaty dotyczą jedynie fragmentu tekstu występującego po średniku (dwukropku). Ilustrują to poniższe dwa przykłady.

Przykład 14.

{liść; ~ wolności} «list rozwodowy»: JSand.

Hasło takie ma następującą reprezentację LMF:

```
<LexicalEntry id="lex.5786">
  <-- INDEX 5261 VERIFY
  <Lemma>
    <FormRepresentation>
      <feat att="writtenForm" val="liść" />
      <feat att="language" val="polh" />
      <feat att="sourceID" val="srpsdp" />
    </FormRepresentation>
  </Lemma>
</LexicalEntry>

<LexicalEntry id="lex.5786:1">
  <Lemma>
    <FormRepresentation>
      <feat att="writtenForm" val="liść wolności" />
      <feat att="language" val="polh" />
      <feat att="sourceID" val="srpsdp:JSand" />
    </FormRepresentation>
  </Lemma>
  <ListOfComponents>
    ...
  </ListOfComponents>
  <Sense>
    <Definition>
      <TextRepresentation>
        <feat att="writtenForm" val="list rozwodowy" />
        <feat att="language" val="pol" />
        <feat att="sourceID" val="srpsdp" />
      </TextRepresentation>
    </Definition>
  </Sense>
</LexicalEntry>
```

Przykład 15.

{przychodzić; ~ o co} «tracić»: O sławę, o majątność, często i o zdrowie przychodzą SGos.

W tym przykładzie mamy do czynienia z rekcją.

```
<LexicalEntry id="lex.12044">
  <-- INDEX 11237 VERIFY
  <Lemma>
    <FormRepresentation>
      <feat att="writtenForm" val="przychodzić" />
      <feat att="language" val="polh" />
      <feat att="sourceID" val="srpsdp" />
    </FormRepresentation>
  </Lemma>
  <SyntacticBehaviour senses="sense.867">
    <feat att="writtenForm" val="przychodzić o co" />
    <feat att="language" val="polh" />
    <feat att="sourceID" val="srpsdp:SGos" />
  </SyntacticBehaviour>
  <Sense id="sense.867">
    <Definition>
      <TextRepresentation>
        <feat att="writtenForm" val="tracić" />
        <feat att="language" val="pol" />
        <feat att="sourceID" val="srpsdp" />
      </TextRepresentation>
    </Definition>
    <Context>
      <TextRepresentation>
        <feat att="writtenForm" val="O sławę, o majątność, często i o
          zdrowie przychodzą" />
        <feat att="language" val="polh" />
        <feat att="sourceID" val="srpsdp:SGos" />
      </TextRepresentation>
    </Context>
  </Sense>
</LexicalEntry>
```

Okolo 5% wszystkich haseł zawiera odnośniki do innych haseł zdefiniowanych w słowniku.

Przykład 16.

{abiecadło} zob. obiecadło

```
<LexicalEntry id="lex.5">
  <-- INDEX 5
  <Lemma>
    <FormRepresentation>
      <feat att="writtenForm" val="abiecadło" />
      <feat att="language" val="polh" />
    </FormRepresentation>
  </Lemma>
</LexicalEntry>
```

```

        <feat att="sourceID" val="srpsdp" />
    </FormRepresentation>
</Lemma>
<RelatedForm targets="lex.7962">
    <FormRepresentation>
        <feat att="writtenForm" val="obiecadio" />
        <feat att="language" val="polh" />
        <feat att="sourceID" val="srpsdp" />
    </FormRepresentation>
</RelatedForm>
</LexicalEntry>

```

Hasło to nie podaje definicji słowa „abiecadio”. Zamiast tego posiada odsyłacz do hasła o identyfikatorze `lex.7962` z przykładu 6. Ponieważ w hasle `lex.7962` wymienione jest słowo „abiecadio” jako forma wariantywna, element `<LexicalEntry id="lex.5">` zostanie usunięty ze słownika.

Teksty haseł, które nie pasują do żadnego z przewidzianych wzorców zostaną przekształcone w elementy `<LexicalEntry>` opatrzone komentarzem `ERROR` i zawierające tylko element `<Lemma>`.

```

< LexicalEntry id="lex.xxxx">
  <-- INDEX yyyy ERROR -->
  <FormRepresentation>
    <feat att="writtenForm" val="text, który może jest lematem" />
    <feat att="language" val="polh" />
    <feat att="sourceID" val="srpsdp" />
  </FormRepresentation>
</LexicalEntry>

```

Bedą one następnie ręcznie uzupełnione.