
MARCIN WOLIŃSKI
MORFEUSZ
REAKTYWACJA
IPI PAN, 7 KWIETNIA 2014

Zespół

Małgorzata Marciniak nadzór ogólny

Marcin Woliński specyfikacja

Michał Lenart implementacja

Jan Daciuk konsultacja automatologiczna

Witold Kieraś reguły łączenia segmentów, testowanie

Jan Szejko Kuźnia, eksport słowników

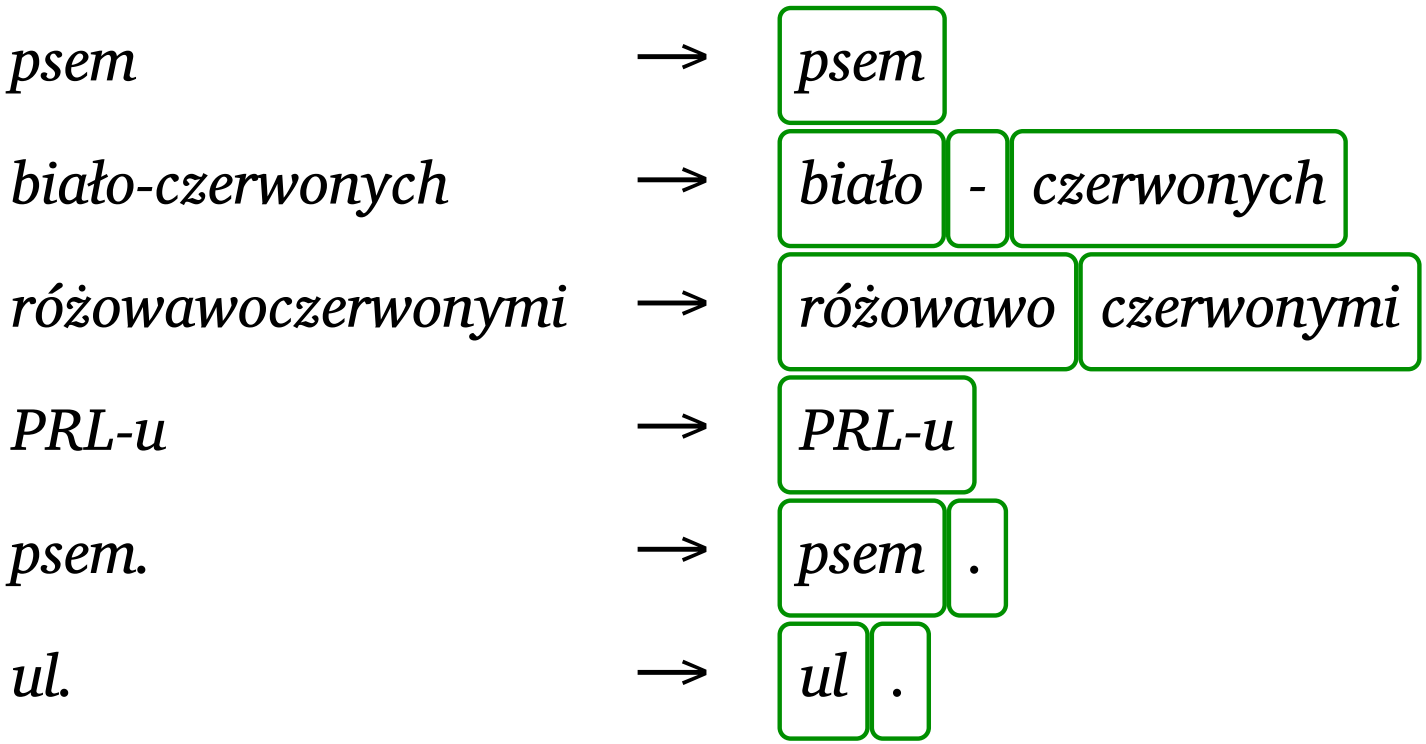
Piotr Rychlik GUI, testowanie

Segmentacja

Słowo maksymalny ciąg znaków nie zawierający odstępu

Segment minimalny odcinek tekstu podlegający interpretacji fleksyjnej

Segmentacja



Segmentacja dla polszczyzny jest uwikłana słownikowo.

Podstawowe pojęcia

Leksem (wyraz słownikowy) abstrakcyjna jednostka języka, zbiór form wyrazowych

Forma (wyrazowa) segment zinterpretowany poprzez przypisanie do leksemu i określenie jego funkcji gramatycznej

Wykładnik (formy) segment reprezentujący ją w tekście

Lemat umowny identyfikator leksemu, tradycyjnie równokształtny z wykładnikiem pewnej jego formy

Technicznie:

Forma trójka $\langle$*wykładnik, lemat, znacznik fleksyjny (tag)*$\rangle$

Leksem zbiór form o tym samym lemacie

Podstawowe pojęcia

Analiza morfologiczna (fleksyjna) identyfikacja wszystkich form wyrazowych, których dany segment może być wykładnikiem

Ujednoznacznianie morfologiczne określenie na podstawie kontekstu, jako którą z możliwych form interpretować dane wystąpienie segmentu

Tagowanie analiza + ujednoznacznienie

Regularność odmiany

TRYB OZNAJMUJĄCY

Czas teraźniejszy^{ndk} / *przyszły*^{dk}

lp

1.os.	**kra**dnę
2.os.	**kra**dniesz
3.os.	**kra**dnie

lm

1.os.	**kra**dniemy
2.os.	**kra**dniecie
3.os.	**kra**dną

Czas przeszły

lp

m	**kra**dł(e)	m	1.os.
ż	**kra**dła	ś	2.os.
n	**kra**dło	∅	3.os.

bezosobnik: **kra**dziono

TRYB ROZKAZUJĄCY

lp

2.os.	**kra**dnij

lm

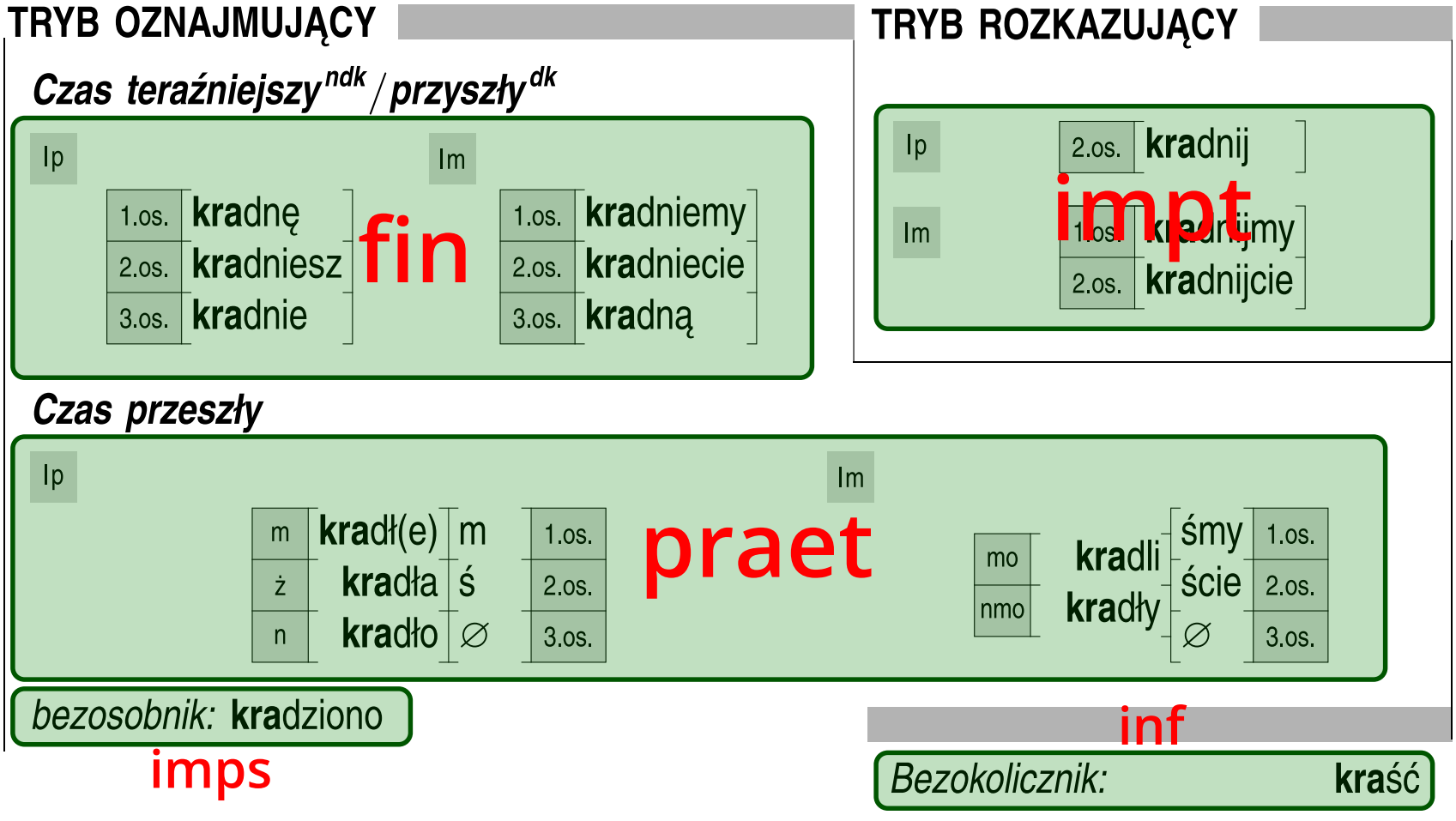
1.os.	**kra**dnijmy
2.os.	**kra**dnijcie

lm

mo	**kra**dli	śmy	1.os.
nmo	**kra**dły	ście	2.os.
		∅	3.os.

Bezokolicznik: **kra**ść

Regularność odmiany



Fleksem podzbiór leksemu (w miarę) jednorodny ze względu na kategorie gramatyczne przysługujące formom

Tagset

Mam	MAMA MAMIĆ MIEĆ	subst:pl:gen:f impt:sg:sec:imperf fin:sg:pri:imperf
próbkę	PRÓBKA	subst:sg:acc:f
analizy	ANALIZA	subst:sg:gen:f subst:pl:nom.acc.voc:f
morfologicznej	MORFOLOGICZNY	adj:sg:gen.dat.loc:f:pos
.	.	interp

Segmentacja form czasu przeszłego

1. *Powiedziała, że**ście** to **czytali**.*
2. *Powiedziała, że to **czytaliście**.*
3. *Powiedziała, żeby**ście** to **czytali**.*
4. **Powiedziała, żeby to **czytaliście**.*

Segmentacja form czasu przeszłego

Morfeusz 1 i 2:

widział	WIDZIEĆ	praet:sg:m1.m2.m3:imperf
em	BYĆ	aglt:sg:pri:imperf:wok

Morfeusz 2 (sterowane opcją):

widziałem	WIDZIEĆ	praet:sg:m1.m2.m3:pri:imperf

Segmentacja form trybu warunkowego

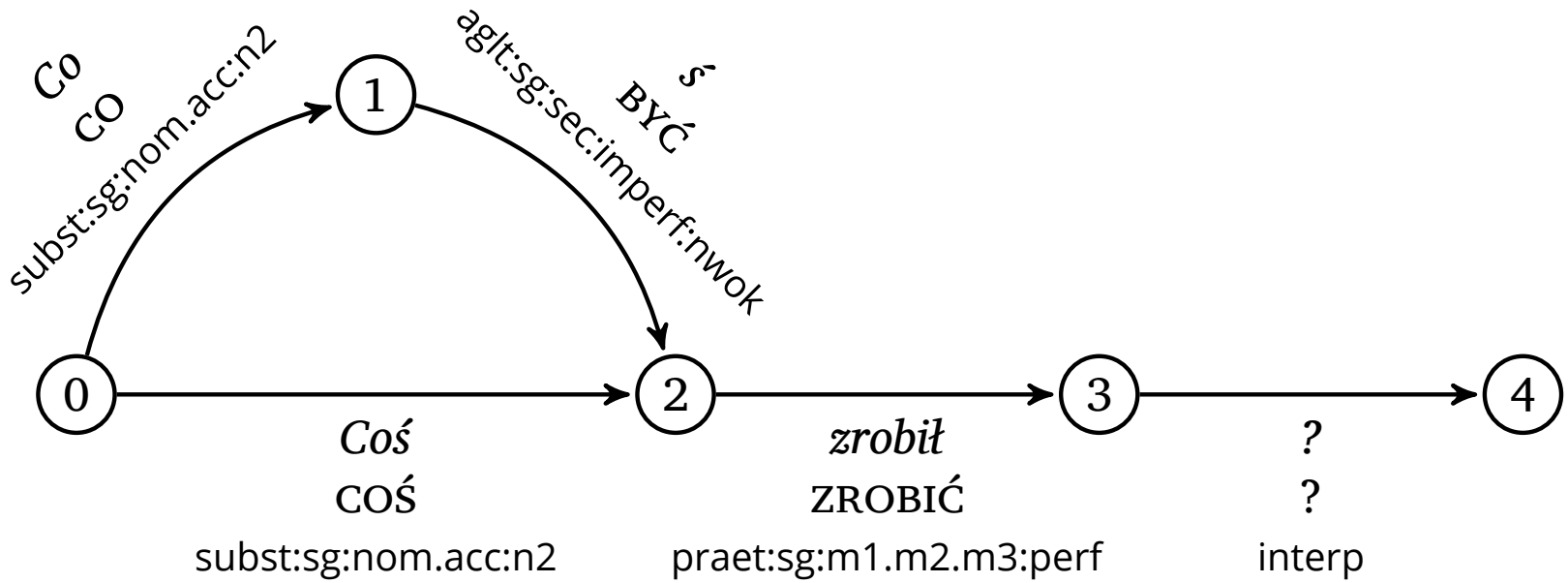
Morfeusz 1 i 2:

widział	WIDZIEĆ	praet:sg:m1.m2.m3:imperf
by	BY	qub
m	BYĆ	aglt:sg:pri:imperf:wok

Morfeusz 2 (nowy znacznik cond):

widziałbym	WIDZIEĆ	cond:sg:m1.m2.m3:pri:imperf

Reprezentacja wyników analizy



Tagsety w okolicy

Następujące tagsety **nie** są identyczne:

- Tagset Morfeusza,
- Tagset Korpusu IPI PAN — w zasadzie przestarzały, mniej szczegółowa klasyfikacja nieodmiennych niż w obecnym Morfeuszu,
- Tagset NKJP — uproszczony system rodzajów, osobna klasa liczebników zbiorowych.

Lematyzacja

Leksem PARA:

- *Uczestnicy tańczą **parami**.*
- *Zatrucie **parami** rtęci jest praktycznie niemożliwe bez jednoczesnego poparzenia.*

Lematyzacja

Leksem PARA:

- *Uczestnicy tańczą **parami**.*
- *Zatrucie **parami** rtęci jest praktycznie niemożliwe bez jednoczesnego poparzenia.*

Leksemy ZAMEK:S1 i ZAMEK:S2:

- *Jakoś odruchowo przekręciła gałkę **zamka**, a potem nacisnęła klamkę.*
- *Na dziedzińcu **zamku** lubelskiego natrafiono na fragmenty konstrukcji zrębowej drewnianej chaty.*

Lematyzacja

- Lematy ok. 10 000 leksemów z SGJP uzupełniono o element ujednoznaczniający.
- Po dwukropku dodano oznaczenie części mowy. Np. leksemy PIEC:S i PIEC:V.
- Jeżeli to nie wystarczyło, dodano oznaczenie cyfrowe, np. ZAMEK:S1 (*zamka*) i ZAMEK:S2 (*zamku*); SŁAĆ:V1 (*ślę*) i SŁAĆ:V2 (*ścielę*).
- Analizator zwraca takie lematy.
- Generator dla argumentu `"piec:s"` zwróci formy odmiany rzeczownika PIEC:S, a dla argumentu `"piec"` — formy zarówno rzeczownika jak i czasownika.

Słowniki

- Morfeusz będzie dystrybuowany z dwoma słownikami: SGJP i Polimorf.
- Kolejne wydania będą generowane automatycznie przez system *Kuźnia* zarządzający pracą nad oboma słownikami.
- Słowniki Morfeusza 2 zostały uzupełnione o informację o byciu nazwą własną i kwalifikatory (np. *daw.*, *przest.*, *rzad.*, *med.*, *chem.*, …).

Postać źródłowa słownika

```
Gdańsk    Gdańsk  subst:sg:acc:m3  geograficzna
Gdańsk    Gdańsk  subst:sg:nom:m3  geograficzna
Gdańska   Gdańsk  subst:sg:gen:m3  geograficzna
Gdański   Gdańsk  subst:pl:nom:m3  geograficzna
Gdańskiem Gdańsk  subst:sg:inst:m3 geograficzna
funkcja   funkcja subst:sg:nom:f   pospolita
funkcjach funkcja subst:pl:loc:f   pospolita
funkcjami funkcja subst:pl:inst:f  pospolita
funkcje   funkcja subst:pl:acc:f   pospolita
funkcje   funkcja subst:pl:nom:f   pospolita
funkcje   funkcja subst:pl:voc:f   pospolita     rzad.
funkcji   funkcja subst:pl:gen:f   pospolita
funkcji   funkcja subst:sg:gen:f   pospolita
funkcjo   funkcja subst:sg:voc:f   pospolita     rzad.
funkcjom  funkcja subst:pl:dat:f   pospolita
funkcyj   funkcja subst:pl:gen:f   pospolita     arch.
```

Kompilowanie słownika

Dane wbudowywane w binarny plik słownikowy Morfeusza:

- słownik źródłowy,
- reguły łączenia segmentów,
- definicja tagsetu.

Łączenie segmentów

- Reguły łączenia segmentów definiują, z jakich segmentów może składać się słowo.
- Każdemu segmentowi przypisujemy typ (na podstawie znacznika i ewentualnie lematu).
- Reguły łączenia są wyrażeniami regularnymi nad typami segmentów.

Przykład reguł łączenia segmentów

```
[tags]
adj_pos adj:%:pos
adv_pos adv:%:pos
adja adja
prefa prefa

[lexemes]
dywiz -:interp

[combinations]
#define adj (adj_pos|adv_pos|adj_com|adv_com|adj_sup)
adj
(adja dywiz)+ adj
adja>+ adj_pos                  !weak
prefa> adj                      !weak
#ifdef permissive
adj (aglsg|aglpl)
#endif
```

Przykład reguł łączenia segmentów

biały

```
[tags]
adj_pos adj:%:pos
adv_pos adv:%:pos
adja adja
prefa prefa

[lexemes]
dywiz -:interp

[combinations]
#define adj (adj_pos|adv_pos|adj_com|adv_com|adj_sup)
adj
(adja dywiz)+ adj
adja>+ adj_pos                  !weak
prefa> adj                      !weak
#ifdef permissive
adj (aglsg|aglpl)
#endif
```

Przykład reguł łączenia segmentów

```
[tags]
adj_pos adj:%:pos
adv_pos adv:%:pos
adja adja
prefa prefa

[lexemes]
dywiz -:interp

[combinations]
#define adj (adj_pos|adv_pos|adj_com|adv_com|adj_sup)
adj
(adja dywiz)+ adj
adja>+ adj_pos                  !weak
prefa> adj                      !weak
#ifdef permissive
adj (aglsg|aglpl)
#endif
```

Przykład reguł łączenia segmentów

średniopienny

```
[tags]
adj_pos adj:%:pos
adv_pos adv:%:pos
adja adja
prefa prefa

[lexemes]
dywiz -:interp

[combinations]
#define adj (adj_pos|adv_pos|adj_com|adv_com|adj_sup)
adj
(adja dywiz)+ adj
adja>+ adj_pos                  !weak
prefa> adj                      !weak
#ifdef permissive
adj (aglsg|aglpl)
#endif
```

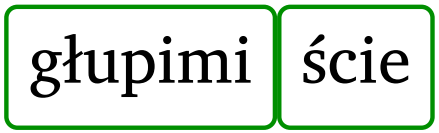
Przykład reguł łączenia segmentów

poliamoryczny

```
[tags]
adj_pos adj:%:pos
adv_pos adv:%:pos
adja adja
prefa prefa

[lexemes]
dywiz -:interp

[combinations]
#define adj (adj_pos|adv_pos|adj_com|adv_com|adj_sup)
adj
(adja dywiz)+ adj
adja>+ adj_pos                  !weak
prefa> adj                      !weak
#ifdef permissive
adj (aglsg|aglpl)
#endif
```

Przykład reguł łączenia segmentów

```
[tags]
adj_pos adj:%:pos
adv_pos adv:%:pos
adja adja
prefa prefa

[lexemes]
dywiz -:interp

[combinations]
#define adj (adj_pos|adv_pos|adj_com|adv_com|adj_sup)
adj
(adja dywiz)+ adj
adja>+ adj_pos                  !weak
prefa> adj                      !weak
#ifdef permissive
adj (aglsg|aglpl)
#endif
```



Warianty segmentacji

- Plik segmentowy deklaruje dostępne warianty segmentacji:

```
[options]
aggl=strict permissive
praet=split composite
```

- Nazwy tych wariantów stają się dostępne jako wartości opcji Morfeusza:
  - **-aggl**
  - **-praet**
- Można zdefiniować kolejne warianty, np. `Lem` i rozpoznawać słowa typu:

***Potrzebowałżebyś**, pytam na koniec, tego strachu wstrętnego i bezsilnej wściekłości.* (Lem, *Przyjaciel Automateusza*)

Wątpliwości segmentacyjne

- Wprowadzamy reguły pozwalające rozpoznawać słowa typu:
  ośmioznakowy,
  studwudziestoipółletni,
  półobywatel,
  dziewięćdziesięciosiedmioipółlatek,
  eurosodoma,
  poliamoryczny.
- Jak je segmentować i lematyzować?

Problem wielkich liter

- Morfeusz ma dwa tryby wrażliwości na wielkie litery:
  - wrażliwy — *Polski* analizowane jako POLSKI i POLSKA; *polski* analizowane tylko jako POLSKI; *andrzej* — `ign`,
  - niewrażliwy — wielkie litery nie wpływają na rozpoznawanie form.
- Być może warto wprowadzić trzeci tryb, różniący się od pierwszego tym, że *andrzej* otrzymuje interpretację ANDRZEJ — bo innej nie ma.

API

- API jest wyrażone w C++, z uproszczonym wariantem w C.
- W programie wielowątkowym należy używać osobnych instancji klasy Morfeusz w poszczególnych wątkach.
- Wykorzystujemy STL. Dwa warianty funkcji analizy i syntezy:
  - iterator, który można poprosić o kolejną formę fleksyjną,
  - wariant zwracający std::vector wszystkich form.

Ładowanie słowników

- Binarne pliki słownikowe są tworzone osobno dla analizatora i generatora.
- API pozwala na określenie nazwy słownika, który ma być używany do analizy i syntezy.
- Próba analizy (syntezy) ze słownikiem `X` spowoduje załadowanie słownika z pliku `morfeusz-X-ana.dict` (`morfeusz-X-gen.dict`) z ustalonej ścieżki.
- Ścieżkę poszukiwania plików słownikowych można ustawić opcją biblioteki.

Biblioteka Morfeusz

- Morfeusz ma postać biblioteki dynamicznej.
- Słowniki są ładowane w czasie pracy, ale jest też możliwość wkompilowania domyślnego słownika w bibliotekę.
- Wątpliwość: jakie warianty dystrybuować?

Dystrybucja programu

- Udostępniamy kod źródłowy i wersje skompilowane dla Linuksa, Mac OS X i Windows; 32- i 64-bitowe.
- Dodatkowe moduły umożliwiają użycie Morfeusza z poziomu Pythona, Perla, Javy i SWI-Prologu.
- Dystrybuujemy w postaci binarnej słownik SGJP i Polimorf.
- Dla mniej technicznych użytkowników przygotowano interfejs graficzny w Javie.

MORFEUSZ REAKTYWACJA

http://sgjp.pl/morfeusz/beta/