Software Engineering Conference Russia 2018
October 12-13, Moscow

Разработка кроссплатформенной библиотеки морфологического анализа текстов на русском языке для использования в промышленных системах

Екатерина Полицына, МАИ
Сергей Полицын, МАИ
Александр Поречный, МАИ

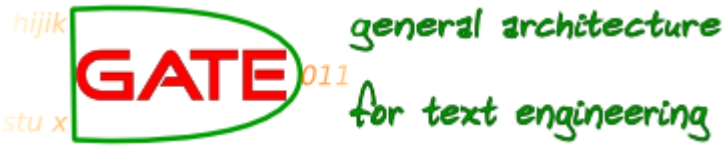
ЗАЧЕМ НУЖНА БИБЛИОТЕКА МОРФОЛОГИИ?

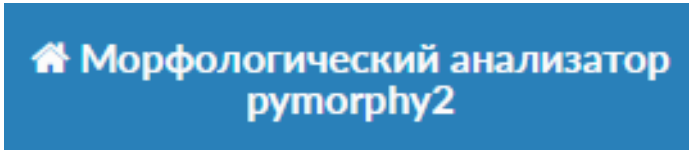
- Информационные системы **используются повсеместно** и оперируют **большими** объемами **данных**, в первую очередь **текстовых**.
- Для **расширения возможностей** систем документооборота, электронной коммерции и др. применяются средства **автоматической** обработки **текстовых** данных.
- Инструменты **морфологического анализа** являются наиболее **востребованными** во многих **промышленных** системах, работающих с **большими объемами** текстовых данных.

ПОЧЕМУ БИБЛИОТЕКА? КАКАЯ?

- Минимизация зависимости от разработчиков инструмента:
  - ✓ Открытая разработка
  - ✓ Не веб-API
- Доступность автономно, т.е. не в составе другой крупной системы
- Кроссплатформенность
- Поддержка русского языка

КАКИЕ ЕСТЬ БИБЛИОТЕКИ?





Russian Morphology for Lucene

CrossMorphy

КАКАЯ НУЖНА БИБЛИОТЕКА?

- Возможность получения морфологических характеристик слова.
- Возможность получения начальной формы слова.
- Возможность получения словоформы по строковому представлению начальной формы и заданным морфологическим характеристикам.
- Кроссплатформенность.
- Высокая производительность.
- Наличие классов для работы с предложениями и словами.

АРХИТЕКТУРА

Файл с морфологическими характеристиками

- Связь словоформ с начальной формой
- Морфологические характеристики словоформ

JMorfSdk

- Получение морфологических характеристик слова
- Получение начальной формы слова
- Получение слова по его начальной форме и морфологическим характеристикам

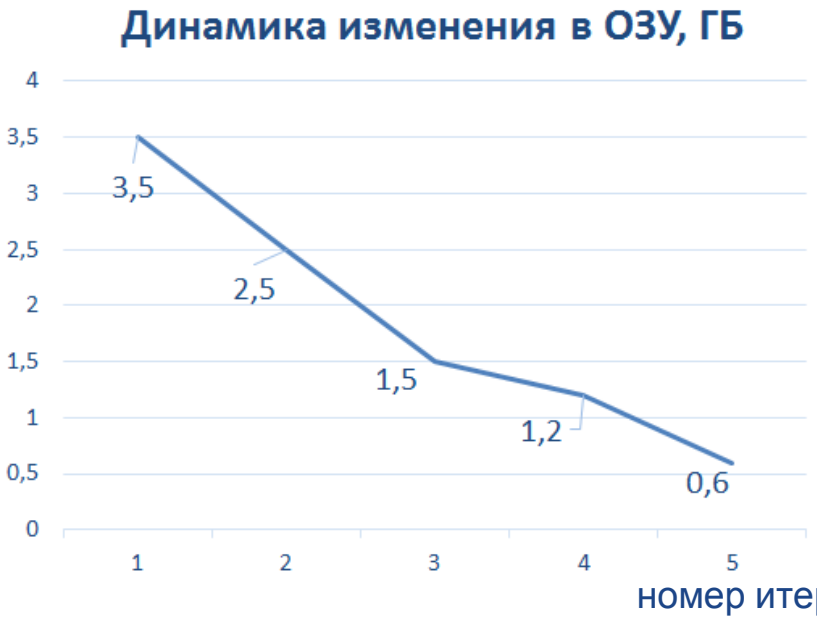
ОПТИМИЗАЦИЯ БИБЛИОТЕКИ

Было:

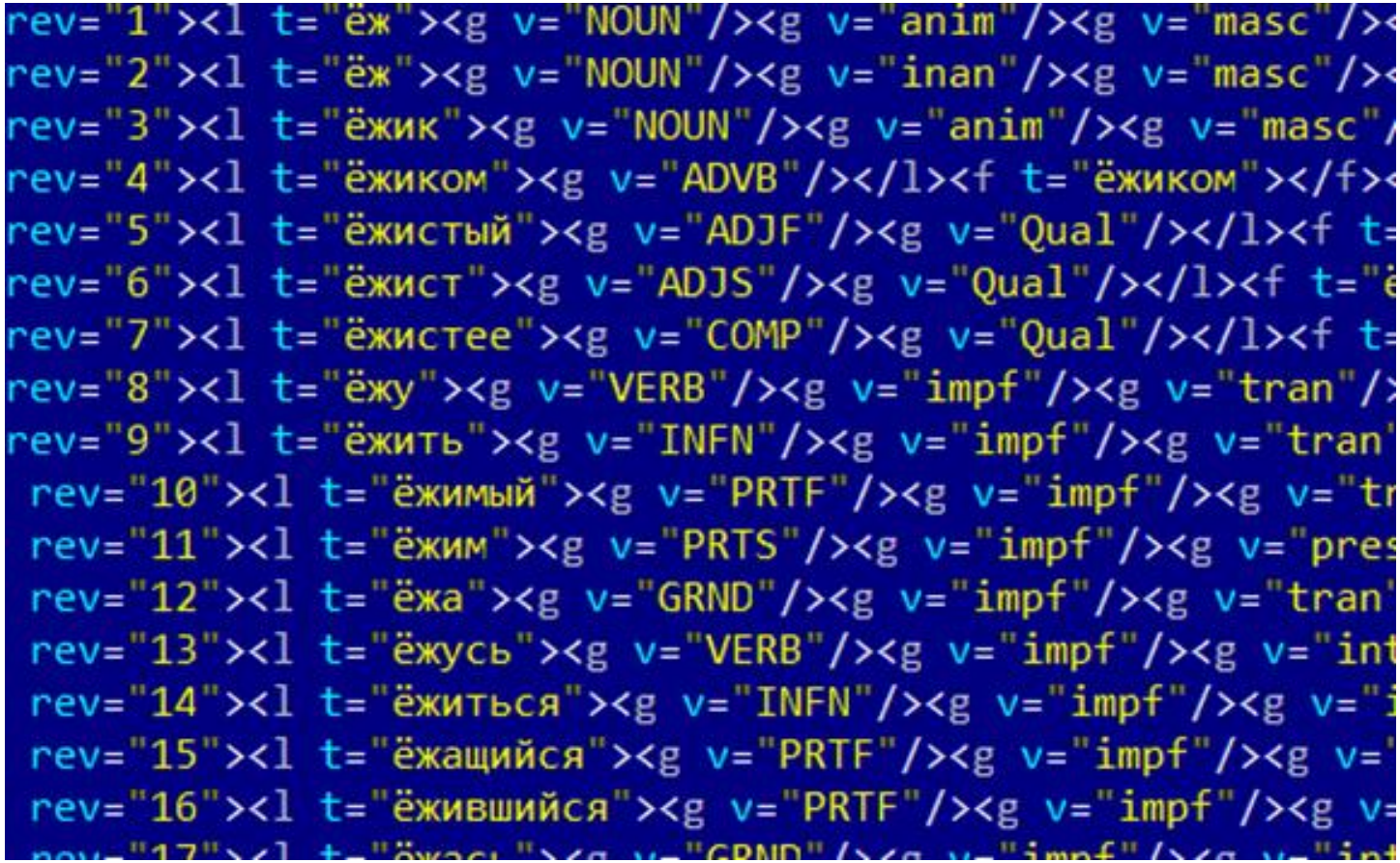
- Словарь в ПЗУ – 500МБ
- Библиотека в ОЗУ 3.5ГБ

Стало:

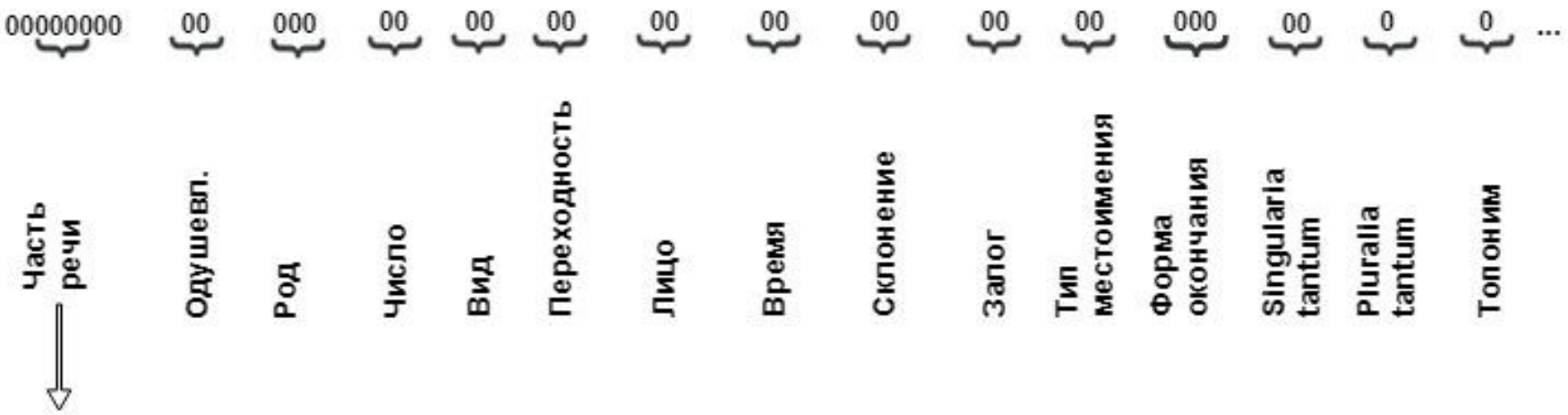
- Словарь в ПЗУ – 35+95МБ
- Библиотека в ОЗУ 550МБ



ИСХОДНАЯ СТРУКТУРА СЛОВАРЯ OpenCorpora

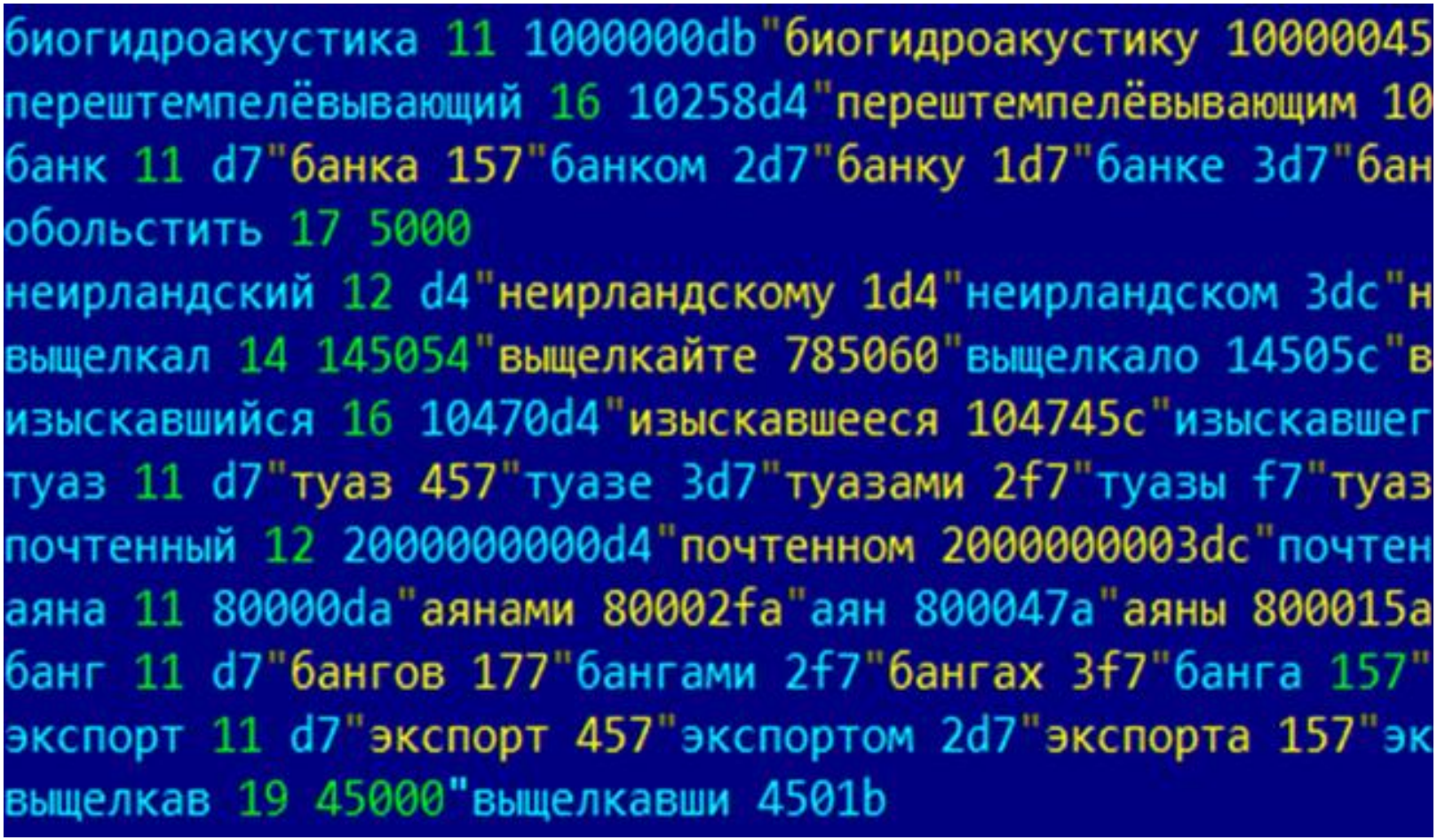


МОРФОЛОГИЧЕСКАЯ ШКАЛА



0001 0001 **Существительное**
0001 0010 **Полное прилагательное**
0001 0011 **Краткое прилагательное**
0001 0100 **Глагол**
0001 0110 **Полное причастие**
0001 1011 **Краткое причастие**
0001 1100 **Деепричастие**
...

ПРОМЕЖУТОЧНАЯ СТРУКТУРА СЛОВАРЯ



СТРУКТУРЫ ДАННЫХ В СЛОВАРЕ

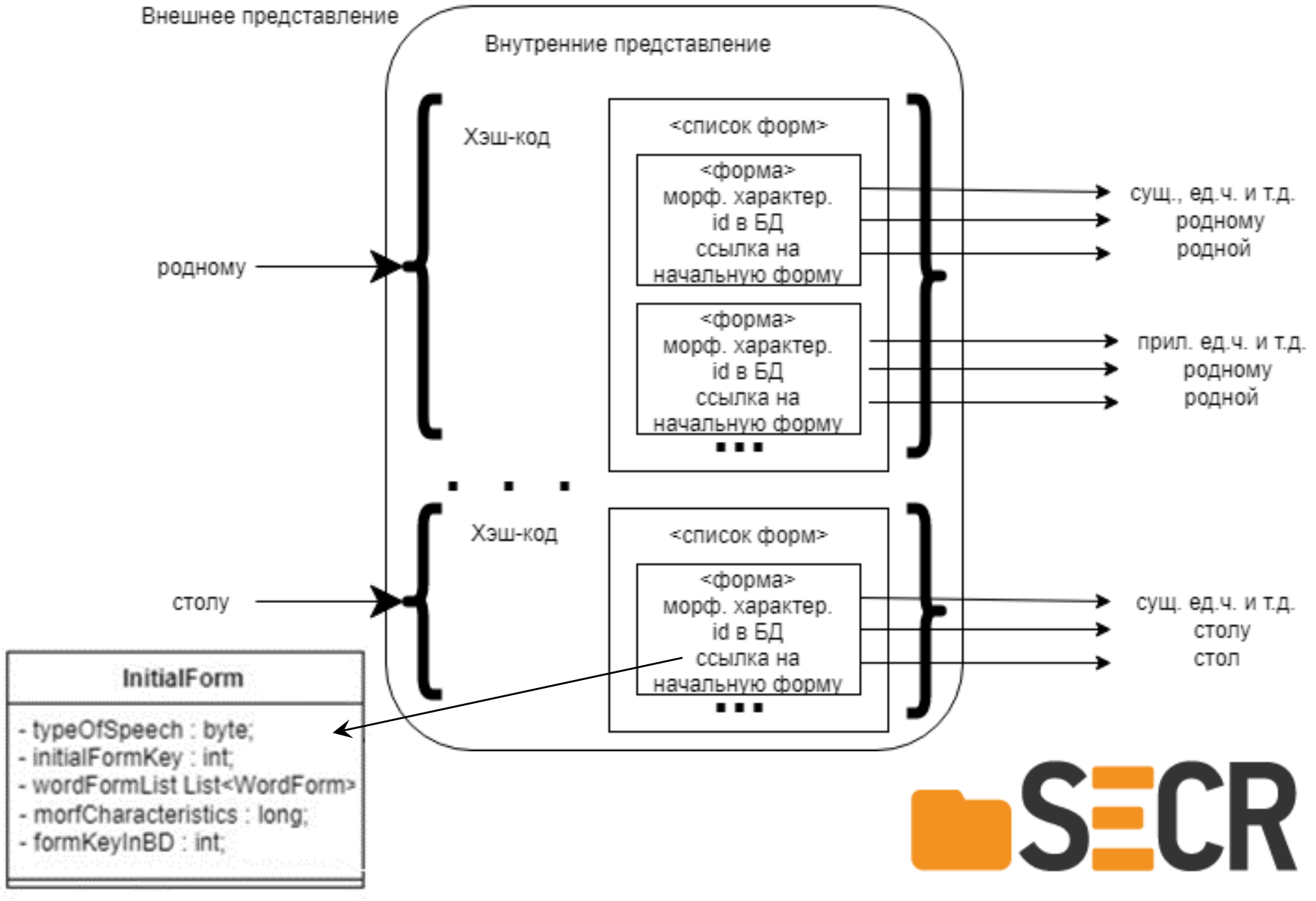
ID	
№ в словаре	Java hash словоформы (старший байт)
3 байта	1 байта

Часть речи (TypeOfSpeech)	
Существительное	0001_0001
Полное прилагательное	0001_0011
Краткое прилагательное	0001_0101
Глагол	0001_0100
Именное местоимение	0001_1101
Наречие	0000_1001
Союз	0000_1000
...	...

Начальная форма (InitialForm)			
CityHash	ID	Часть речи	Морфологические характеристики
4 байта	4 байта	1 байт	8 байт

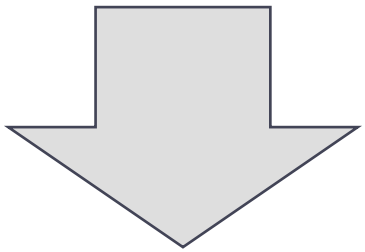
Морфологические характеристики (MorphologicalCharacteristics)	
Одушевленное	0000_..._0000_0000_0010
Единственное число	0000_..._0000_0000_0010
Множественное число	0000_..._0000_0000_0011
...	...
Именительный падеж	0000_..._0000_1000_0000
Родительный падеж	0000_..._0010_1000_0000
...	...

ОПТИМИЗАЦИЯ СТРУКТУР ДАННЫХ



ПОИСК СЛОВОФОРМЫ

1. Получение хэш-кода по алгоритму Google CityHash64 (ключ в HashMap).
2. Получение хэш-кода строки по встроенному алгоритму Java (*.hashCode()).*
3. Пересечение результатов поиска по этим двум хэш-кодам.



4. Разрешение коллизий.

СТРУКТУРА ЗАПИСИ СЛОВАРЯ

Лемма 1

Лемма 2

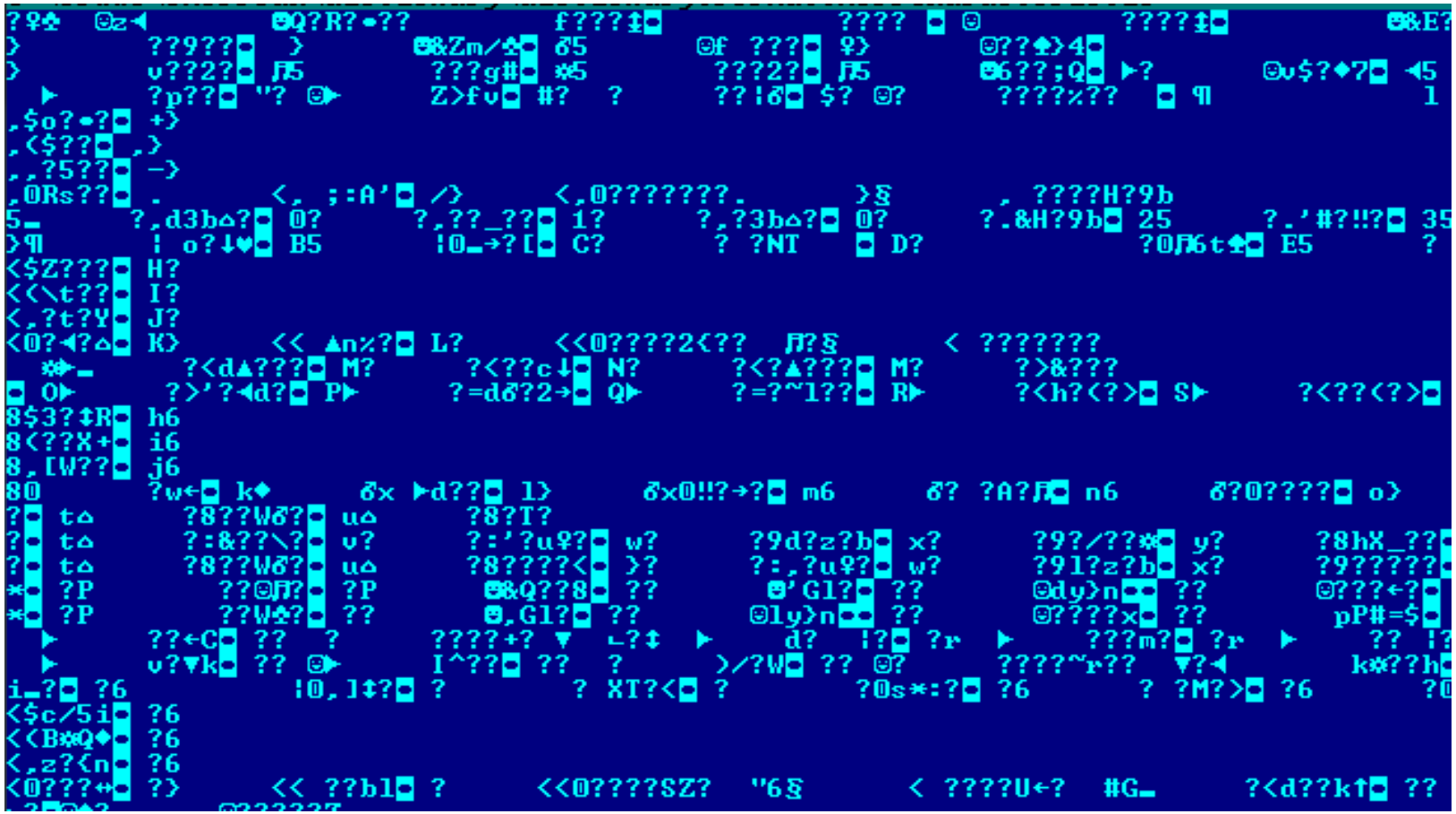
17 байт	16 байт	16 байт	...	16 байт	4 байта	17 байт	16 байт	...
InitialForm	WordForm	WordForm	...	WordForm	ControlValue	InitialForm	WordForm	...

InitialForm			
CityHash	ID	TypeOfSpeech	MorphologicalCharacteristics
4 байта	4 байта	1 байт	8 байт

WordForm		
CityHash	ID	MorphologicalCharacteristics
4 байта	4 байта	8 байт

ControlValue
1111_1111 x 4 байта

ИТОГОВАЯ СТРУКТУРА СЛОВАРЯ



ID	Строковое представление слова
231512	биогидроакустика

ИТОГОВАЯ АРХИТЕКТУРА

MorphologicalStructures

- Конвертация морфологического словаря в структуру, применяемую в JMorfSdk
- Работа с БД, в которой хранятся текстовые формы словоформ

JMorfSdk

- Получение морфологических характеристик слова
- Получение начальной формы слова
- Получение слова по его начальной форме и морфологическим характеристикам

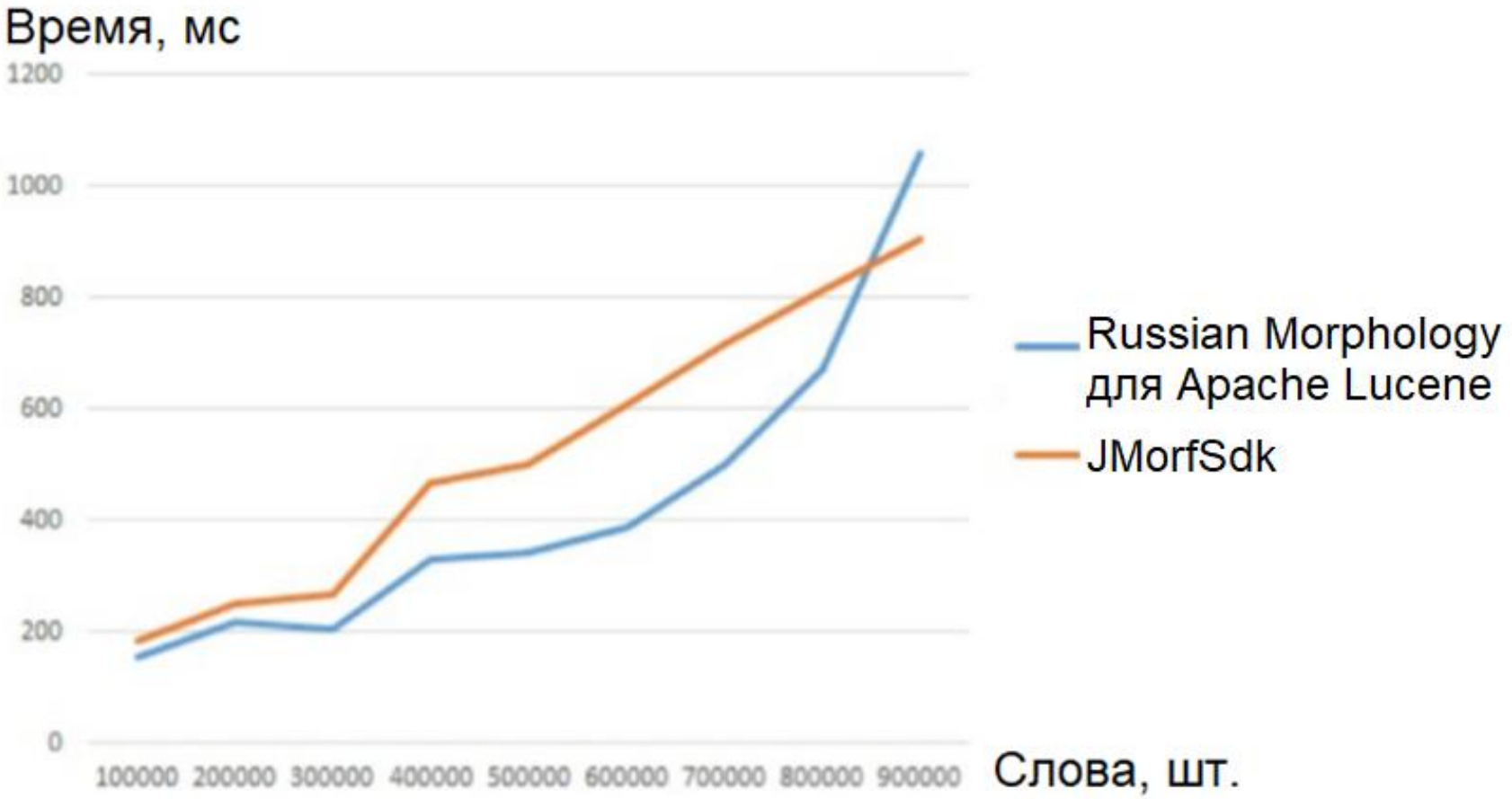
Бинарный файл с морфологическими характеристиками

- Связь словоформ с начальной формой
- Морфологические характеристики словоформ в бинарном виде
- Содержит идентификаторы словоформ для БД

БД (SQLite)

- Две БД по содержанию: только начальные формы и производные от них словоформы
- Текстовое представление словоформ, доступное по идентификатору

СРАВНЕНИЕ С JAVA RUSSIAN MORPHOLOGY ДЛЯ APACHE LUCENE



ФИЛЬТРАЦИЯ СЛОВ ПО МОРФОЛОГИЧЕСКИМ ХАРАКТЕРИСТИКАМ

```
List<String> words = Arrays.asList("красного", "красный", "морозное", "солнечная",
                                   "бежать", "доска", "топор", "шла");

for (String word : words) {
    jMorfSdk.getAllCharacteristicsOfForm(word).forEach(form -> {
        if (form.getTheMorfCharacteristics(MorfologyParameters.Gender.class)
            == MorfologyParameters.Gender.FEMININ) {
            System.out.println(form + " - " + word);
        }
    });
}
```

```
initialFormString = солнечный, typeOfSpeech = 18, morfCharacteristics = 4200 - солнечная
initialFormString = доска, typeOfSpeech = 17, morfCharacteristics = 107 - доска
initialFormString = иду, typeOfSpeech = 20, morfCharacteristics = 670760 - шла
```

ПОЛУЧЕНИЕ ЗАДАННОЙ ФОРМЫ СЛОВА

```
sdk.getAllCharacteristicsOfForm("дорогой").forEach((form) -> {
    if (form.getTheMorfCharacteristics(MorfologyParameters.Case.IDENTIFIER) ==
            MorfologyParameters.Case.GENITIVE){
        System.out.println(form);
    }
});
```

```
initialFormString = дорогой, typeOfSpeech = 18, morfCharacteristics = 4264
```

```
sdk.getAllCharacteristicsOfForm("дорогой").forEach((form) -> {
    if (form.getTypeOfSpeech() == MorfologyParameters.TypeOfSpeech.NOUN){
        System.out.println(form);
    }
});
```

```
initialFormString = дорога, typeOfSpeech = 17, morfCharacteristics = 363
```

ПОЛУЧЕНИЕ СЛОВА С ЗАДАНЫМИ ХАРАКТЕРИСТИКАМИ

```
jMorfSdk.getDerivativeForm("мыло",
        TypeOfSpeech.NOUN,
        Numbers.SINGULAR)
            .forEach((wordString) -> {
              System.out.println(wordString);
            });
```

мылом
мыло
мыле
мыла
мылу

ГДЕ ИСПОЛЬЗОВАЛИ?

- В **системе классификации текстов** на основе ключевых слов и словосочетаний.
- При решении задачи **выделения именованных сущностей.**
- Для **восстановления полных форм** слов при работе с сокращениями.
- В **исследовательских задачах** компьютерной лингвистики: морфологический этап анализа - один из основных, необходим для большинства задач.

СОЗДАННАЯ БИБЛИОТЕКА

Реализованная кроссплатформенная библиотека JMorfSdk имеет:

1) Режим анализа: определение морфологических характеристик слова и получение начальной формы со сложностью **O(1)**.

2) Режим генерации: получение слова по строковому представлению слова и набору морфологических характеристик.

https://github.com/jalexpr/JMorfSdk

```
<dependency>
    <groupId>ru.textanalysis.tfwwt</groupId>
    <artifactId>jmorfsdk</artifactId>
    <version>2.10.5</version>
</dependency>
```

СПАСИБО ЗА ВНИМАНИЕ!

Ekaterina Politsyna
Sergey Politsyn
Alexander Porechny

http://textanalysis.ru/
tasystem@yandex.ru