

# API для программного продукта "Приведение слова к словарной форме" (CWFAPI - Common Iex Word Form Application Program Interface)

## Содержание

Программный модуль предназначен для определения словарной формы произвольного русского слова. Реализован в виде пары "лингвистическое ядро - словари" (DLL-файл и LEX-файлы). Общается с вызывающим его приложением через специальный программный интерфейс WFAPI, описываемом в настоящем документе.

1.	Инициализация .....	1
2.	Открытие основного словаря .....	1
3.	Открытие словаря пользователя .....	2
4.	Запрос на словарную форму .....	2
5.	Получение хэш-кода .....	4
6.	Запрос на все словоформы .....	4
7.	Закрытие словаря пользователя .....	5
8.	Закрытие основного словаря .....	5
9.	Завершение сеанса .....	5

## 1. Инициализация

```
WFEC CWFAPI WFINit ( WFID FAR * lpwfid );
```

### Параметры

*lpwfid* - указатель, используемый при всех следующих вызовах лингвистического модуля;

### Описание

Отправная точка сеанса работы WF. После успешной инициализации должна быть вызвана программа WFOpenMdt(), после которой могут вызываться лингвистические программы. Если WFINit() отработает неудачно (вернет не 0), нельзя будет вызвать никакие другие программы, кроме WFINit(), так как для других программ требуется правильный дескриптор памяти. WFINit() инициализирует структуры и ресурсы, необходимые для последующих вызовов WF.

Указатель *lpwfid* должен быть дескриптором памяти, отведенной под переменные. Указатель *lpwfid* должен быть установлен в *wfidNil*, если случится ошибка.

### Возвраты

Возвращает *wfecNone* при успешной инициализации и обновляет параметр *lpwfid*. В противном случае возвращает WFEC, идентифицирующий проблему. Возможные значения WFEC: *wfecNone* (=0 - нет ошибок), *wfecModuleInUse*, *wfecOOM* (out of memory).

## 2. Открытие основного словаря

```
WFEC CWFAPI WFOpenMdt ( WFID          wfid,  
                        LPSPATH lpspathMdt  
                        // Указатель на путь к основному словарю;  
                        );
```

**Параметры**

*wfid* - идентификатор, возвращаемый WFInit();

*lpspathMdt* - Указатель на полный путь к основному словарю (ANSI).

**Описание**

Эта программа регистрирует указанный основной словарь и позволяет пользоваться им в последующей работе WF.

Основной словарь открывается и проверяется, но не обязательно загружается. Эта программа должна быть вызвана после WFInit() и до других вызовов WF.

**Возвраты**

Возвращает wfecNone при успехе. В противном случае возвращает WFEC, идентифицирующий проблему.

Возможные значения WFEC: wfecNone (=0 - нет ошибок), wfecModuleInUse, wfecHaveMainDict (уже открыт основной словарь), wfecBadMainDict (- плохой основной словарь), wfecUnknown (- неизвестная ошибка).

**3. Открытие словаря пользователя**

```
WFEC CWFAPI WFOpenUdr ( WFID      wfid,
                       LPSTR     lpspathUdr
                       // Указатель на путь к словарю пользователя;
                       );
```

**Параметры**

*wfid* - идентификатор, возвращаемый WFInit();

*lpspathUdr* - Указатель на полный путь к словарю пользователя (ANSI).

**Описание**

Эта программа открывает указанный морфологический словарь пользователя и позволяет пользоваться им в последующей работе WF. Словарь пользователя открывается, проверяется и загружается. Эта программа должна быть вызвана после WFInit() и WFOpenMdt() и до других вызовов WF.

**Возвраты**

Возвращает wfecNone при успехе. В противном случае возвращает WFEC, идентифицирующий проблему.

Возможные значения WFEC: wfecNone (=0 - нет ошибок), wfecModuleInUse, wfecNoMainDict (основной словарь еще не загружен), wfecUnknown (неизвестная ошибка), wfecBadUserDict (испорчен словарь пользователя).

**4. Запрос на словарную форму**

```
typedef struct {
    LPSTR      lpszLexWordForm;
    enum WFPOS iPartOfSpeech;
    DWORD      dwHighWordId;
    DWORD      dwLowWordId;
} HOM;

WFEC CWFAPI WFLexForm ( WFID      wfid,
                       LPCSTR    lpszInputWord,
                       LPDWORD    lpdwMorph,
                       LPWORD     lpwWordHomNum,
                       HOM        *lpHom
                       );
```

**Параметры:**

*wfid* - идентификатор, возвращаемый WFInit();

*lpszInputWord* - указатель на исходное слово;

*lpdwMorph* – указатель на 32-битное число, может принимать значение 1 или 0. Если = 1, то ищет словарную форму в словаре. Если = 0, то в полях структуры **HOM** dwHighWordId и dwLowWordId возвращает значение эквивалентное вызову функции WFCCommonHashCode, остальные поля структуры значения не имеют.

Если слово найдено в словаре (*dwMorph=1* на входе), то *dwMorph* сохраняет значение равное 1. Если не найдено, то значение становится равным 0, а результат поиска эквивалентен запросу со значением *dwMorph=0*. *lpwWordHomNum* - входной/выходной параметр. На входе – максимальное кол-во словарных статей (структур НОМ), которое может быть сформировано на выходе для данного слова. В большинстве случаев будет найдена только одна статья. Для того, чтобы заведомо получать всю информацию из словаря, этот параметр д.б. больше или равен MAX\_OF\_HOM. На выходе – реально найденное количество структур НОМ для данного входного слова;

*lpHom* - указатель на выходной массив структур НОМ, содержащих информацию о каждом омониме, объясняющем исходное слово. В каждой структуре НОМ содержится:

*lpszLexWordForm* - указатель на словарную форму (память не менее MAX\_WORD\_LEN байт заказывает и освобождает вызывающая программа);

*iPartOfSpeech* - идентификатор части речи (см. файл swfapi.h);

```
enum WFPOS {
    wfnoun = 0,      // Существительное
    wfadject,       // Прилагательное
    wfverb,         // Глагол
    wfauxil,        // Неизменяемое (кроме наречия, союза, частицы и т.д.)
    wfprepos,       // Предлог
    wfpronoun,     // Местоимение
    wfnumber,       // Числительное
    wfabbr_with_period, // Сокращение с точкой
    wfparticiple,   // Причастие
    wfinterjection, // Междометие
    wfadverb,       // Наречие
    wfparenthesis,  // Вводное слово
    wfconjunction,  // Союз
    wfparticle,     // Частица
    wfinterrogative // Вопросительное слово
};
```

*dwHighWordId* и *dwLowWordId* - старшая и младшая половины идентификационного номера слова. Реально используется только *dwHighWordId*. Младшая половина *DwLowWordId == 0* и зарезервирована для возможного наращивания идентификационного номера.

## Описание

Эта программа ищет по основному и пользовательскому словарю все статьи, объясняющие исходное слово.

Если слова нет в словаре, возвращает wfecNotInDict. (После такого возврата можно вызывать модуль обучения и добавить слово в словарь пользователя).

По найденным морфологическим статьям формируются словарные формы исходного слова и части речи, присваивается четырехбайтовый идентификационный номер (Младшая половина *DwLowWordId == 0*).

Замечание о больших/малых буквах:

В словарной форме капитализация устанавливается как в словаре, независимо от капитализации исходного слова, например для слов “Москва”, “МОСКВА” и даже “москва” будет выдана словарная форма “Москва”.

Замечание о работе со словами через дефис:

- 1 Слова с дефисом подаются целиком, так что вызывающая программа **не должна** сама заботиться о правильном разбиении слов через дефис;
- 2 Если слово с дефисом есть в словаре целиком, то оно и обрабатывается целиком, например, “Рио-де-Жанейро”, “кирпично-красный”;
- 3 Для слов с дефисным написанием частиц частица отсекается и слово обрабатывается без частицы, например, “платить-таки”;
- 4 Существительные через дефис с обеими изменяемыми частями отсутствуют в словаре целиком, поэтому будут сформированы структуры НОМ как для первой, так и для второй части. Например, для слова “женщина-врач” будут сформированы 2 структуры НОМ, одна для слова “женщина” и вторая для слова “врач”.
- 5 Прилагательные через дефис могут отсутствовать в словаре целиком, но первая часть может присутствовать как краткое, а вторая как полное прилагательное. В этом случае сформируются структуры

НОМ как для отдельных частей, так и для синтезированного слова целиком. Например, для слова “бледно-научный” будут сформированы структуры НОМ для слов “бледный”, “научный” и “бледно-научный”

Замечание о причастных формах:  
 Для глаголов синтезируются все причастия в форме именительного падежа мужского рода единственного числа. Если функция используется для индексирования текстовых баз, причастия можно не включать в индекс.

**Возвраты**

Возвращает wfecNone при успехе. В противном случае возвращает WFEC, идентифицирующий проблему. Возможные значения WFEC: wfecNone (=0 - нет ошибок), wfecNotInDictInput (нет в словаре), wfecNotRussianWord (не русское слово), wfecNoMainDict (основной словарь еще не загружен), wfecUnknown (неизвестная ошибка), wfecOOM (out of memory).

**5. Получение хэш-кода**

```
WFEC CWFAPI WFCommonHashCode (WFID wfid,
                             LPCSTR lpszInputWord,
                             DWORD *dwHighWordId,
                             DWORD *dwLowWordId);
```

**Параметры:**

wfid - идентификатор, возвращаемый WFInit();  
 lpszInputWord - указатель на исходное слово;  
 dwHighWordId - выходной параметр – хэш-код, приписываемый исходному слову;  
 dwLowWordId - выходной параметр, в данный момент не используется.

**Описание**

Эта функция генерирует хэш-код для указанного слова вне зависимости от того, есть оно в словаре ОРФО или нет.

**Замечание**

При вызове этой функции следует передавать исходное слово в нижнем регистре для возможности поиска слов, отличающихся только капитализацией.

**Функция генерирует хэш-код без учета морфологии!** Т.е. все омонимы будут иметь один и тот же хэш-код.

**Возвраты**

Возвращает wfecNone при успехе. В противном случае возвращает WFEC, идентифицирующий проблему. Возможные значения WFEC: wfecNone (=0 - нет ошибок), wfecUnknown (неизвестная ошибка), wfecOOM (out of memory).

**6. Запрос на все словоформы**

```
WFEC CWFAPI WFAAllForms ( WFID wfid,
                          LPCSTR lpszInputWord,
                          enum WFPOS iPartOfSpeech,
                          LPSTR lprgszOutputWords,
                          BOOL bDoFullOutput
                          );
```

**Параметры**

wfid - идентификатор, возвращаемый WFInit();  
 lpszInputWord - указатель на исходное слово;  
 iPartOfSpeech – номер части речи;  
 lprgszOutputWords - указатель на все формы слова, синтезируемые WFAAllForms (память не менее MAX\_ALL\_FORMS\_LEN байт заказывает и освобождает вызывающая программа). Словоформы перечисляются через \0. В конце списка стоит \0\0;  
 bDoFullOutput – режим выдачи словоформ:  
     bDoFullOutput = TRUE – выдавать все формы всех омонимов, включая повторы форм и отсутствующие формы  
     bDoFullOutput = FALSE – выдать список без повторов и отсутствующих форм

**Описание**

Эта программа ищет по основному и пользовательскому словарю все статьи, объясняющие исходное слово в соответствии с специфицированной частью речи. Если слова нет в словаре, возвращает wfecNotInDict. (После такого возврата можно вызывать модуль обучения и добавить слово в словарь пользователя). По найденным морфологическим статьям формируются все формы исходного слова.

**Возвраты**

Возвращает wfecNone при успехе. В противном случае возвращает WFEC, идентифицирующий проблему. Возможные значения WFEC: wfecNone (=0 - нет ошибок), wfecNotInDictInput (нет в словаре), wfecNotRussianWord (не русское слово), wfecUncorresponded (специфицированная часть речи не соответствует слову), wfecNoMainDict (основной словарь еще не загружен), wfecUnknown (неизвестная ошибка), wfecOOM (out of memory).

## 7. Закрытие словаря пользователя

```
WFEC CWFAPI WFCloseUdr (WFID wfid);
```

**Параметры**

wfid - идентификатор, возвращаемый WFInit();

**Описание**

Закрывает морфологический словарь пользователя, открытый WFOpenUdr () и делает его недоступным для WF. WFCloseUdr() должна быть вызвана до вызова WFCloseMdt() (т.е. пока открыт основной словарь).

**Возвраты**

Возвращает wfecNone при успехе. В противном случае возвращает WFEC, идентифицирующий проблему. Возможные значения WFEC: wfecNone (нет ошибок), wfecModuleInUse, wfecNoMainDict (словарь еще не загружен), wfecUnknown (неизвестная ошибка).

## 8. Закрытие основного словаря

```
WFEC CWFAPI WFCloseMdt (WFID wfid);
```

**Параметры**

wfid - идентификатор, возвращаемый WFInit();

**Описание**

Закрывает основной словарь и делает его недоступным для WF. Функции WF не могут быть использованы, пока WFOpenMdt() снова не откроет какой-либо словарь. WFCloseMdt() должна быть вызвана до вызова WFterminate().

**Возвраты**

Возвращает wfecNone при успехе. В противном случае возвращает WFEC (WF Error Code), идентифицирующий проблему. Возможные значения WFEC: wfecNone (=0 - нет ошибок), wfecModuleInUse, wfecUnknown (неизвестная ошибка).

## 9. Завершение сеанса

```
WFEC CWFAPI WFterminate ( WFID wfid );
```