



ЗАО СофтПро, www.cboss.ru

УТВЕРЖДЕН
42 5100 5-003-45162042-00 И6-ЛУ

CBOSSsms 3.0

СИСТЕМА КОРОТКИХ СООБЩЕНИЙ

ТРЕБОВАНИЯ К БАЗЕ ДАННЫХ ПРОВАЙДЕРА ИНФОРМАЦИИ

Редакция 3.0.0.3

42 5100 5-003-45162042-00 И6

СофтПро
2001

CBOSSsms 3.0. Требования к базе данных провайдера информации. Редакция 3.0.0.3

42 5100 5-003-45162042-00 И6

Авторы: Заикин А.В., Лысенков В.С

Документ содержит требования и инструкции, выполнение которых необходимо для обеспечения провайдером информации доступа к своей базе данных со стороны пользователей системы коротких сообщений CBOSSsms.

Copyright © SoftPro 2001. Все права защищены.

Настоящий документ содержит информацию, актуальную на момент его составления. СофтПро не гарантирует отсутствие ошибок в данном документе. СофтПро оставляет за собой право вносить изменения в документ без предварительного уведомления.

Никакая часть настоящего документа не может быть воспроизведена или передана в какой бы то ни было форме и какими бы то ни было средствами, без письменного на то разрешения СофтПро.

СофтПро не гарантирует, что специфицированное в настоящем документе программное обеспечение будет удовлетворять всем требованиям, которые могут быть к нему предъявлены, и что программное обеспечение будет работать в произвольно выбранных условиях, а также что его работа будет непрерывной и что оно не содержит в себе программных ошибок.

СофтПро не гарантирует работоспособность нелегально полученного программного обеспечения. Нелегальное использование программного обеспечения и документации на него преследуется по закону.

CBOSS, COOBC, COOCB, CBOSSacc, CBOSSacr, CBOSSdss, CBOSSics, CBOSSisp, CBOSS prepaid, CBOSSsms, CBOSSssc, CBOSStmn, CBOSSvmail, CBOSSvote являются зарегистрированными товарными знаками.

SoftPro, СофтПро являются товарными знаками.

ORACLE, Developer/2000, Oracle Book, Oracle Forms, Oracle Graphics, Oracle Help, Oracle*Mail, Oracle Reports, PL/SQL, Pro*C, SQL*Net, SQL*Plus являются товарными знаками **Oracle Corporation**.

MS, Microsoft являются зарегистрированными товарными знаками **Microsoft Corporation**.

Windows, Windows 95, Windows 2000 являются товарными знаками **Microsoft Corporation**.

Все другие товарные знаки и зарегистрированные товарные знаки могут являться собственностью соответствующих компаний.

Содержание

Отправьте нам свои комментарии	v
Введение	vii
Редакции	ix
1 Общие положения	1-1
2 Информирование по запросу	2-1
3 Автоматическое информирование	3-1

Отправьте нам свои комментарии

CBOSSsms 3.0. Требования к базе данных провайдера информации. Редакция 3.0.0.3

42 5100 5-003-45162042-00 И6

Мы будем рады Вашим предложениям и замечаниям по качеству исполнения и полноте содержания данной публикации. Ваш вклад очень важен для её совершенствования. Сообщите нам, если Вы

- нашли ошибки;
- считаете, что информация изложена недостаточно полно;
- имеете необходимость в дополнительной информации (укажите, в какой именно);
- обнаружили, что материал изложен не вполне ясно и для его понимания Вам требуются дополнительные разъяснения и примеры;
- желаете отметить те или иные особенно понравившиеся Вам черты данной публикации и хотите, чтобы они были сохранены в будущем.

Пожалуйста, оформите Ваши отзывы и предложения в письменном виде (при этом не забывайте указывать название, десятичный номер и номер редакции документа, а также, по возможности, главу, секцию и страницу, к которым относятся Ваши замечания), а затем отправьте их нам одним из указанных ниже способов:

- посредством рабочего места **Замечания**, которое вызывается из любого программно-функционального модуля CBOSS
- абонентский запрос в службу технической поддержки СофтПро
- электронная почта-email@cboss.ru
- факс-+7 (095) 974-6388
- почта:
СофтПро
Отдел документирования
Россия, 117630 Москва, Старокалужское ш., д. 62, стр. 1.

Введение

Данный документ содержит требования и инструкции, выполнение которых необходимо для обеспечения провайдером информации доступа к своей базе данных со стороны пользователей системы коротких сообщений CBOSSms.

На кого ориентирован документ

Документ предназначен для использования опытными администраторами и разработчикам баз данных провайдера информации.

Как организован данный документ

Документ состоит из следующих глав:

Глава 1. “Общие положения”.

Содержит самые общие сведения о системе.

Глава 2. “Информирование по запросу”.

Содержит подробное описание технологии получения данных от провайдера информации по запросу пользователя услуги коротких сообщений.

Глава 3. “Автоматическое информирование”.

Содержит подробное описание технологии автоматической (без запроса) рассылки данных от провайдера информации для пользователей услуги коротких сообщений.

Раздел “Редакции”.

Отражает историю развития документа.

Как пользоваться данным документом

Документ может быть использован как для последовательного чтения, так и в режиме справки.

Типографские соглашения, используемые в данном документе

Курсивом в тексте выделяются термины и понятия, но только там, где даётся их толкование. Также курсивом в тексте документа даются ссылки на другие документы и публикации. Тем же способом в тексте выделяются слова и словосочетания, которым в контексте изложения придаётся особое, отличное от обычного, значение.

Крупным жирным шрифтом в тексте выделяются названия элементов графического интерфейса пользователя (окон, кнопок, полей ввода и т.п.). Такое выделение используется только при (первоначальном) описании элементов. В прочих местах текста их названия даются **обычным жирным** шрифтом.

В кавычки заключаются значения констант, параметров, переменных, а также названия глав документа, когда на них делаются ссылки в тексте.

Документы, рекомендуемые для предварительного прочтения

CBOSSsms 3.0. Общее описание системы – М: СофтПро, 2000.

CBOSSsms 3.0. Руководство администратора – М: СофтПро, 2000.

CBOSSsms 3.0. Руководство пользователя – М: СофтПро, 2000.

Редакции

Редакция 3.0.0.3

от 19.10.2001 г.

- Добавлено замечание о возможности использования **cbSDP** вместо **Dbdrv**.

Редакция 3.0.0.2

от 27.04.2001 г.

- Изменено оформление и месторасположение сведений о редакциях документа.

Редакция 3.0.0.1

от 12.01.2001 г.

- Изменён стиль оформления документа.
- Исправлены обнаруженные неточности.

Редакция 3.0.0.0

от 21.08.2000 г.

- Новый документ.

1 Общие положения

Программно-аппаратный комплекс CBOSSsms обладает широким спектром возможностей, к которым относятся обработка запросов пользователей CBOSSsms, направленных к базе данных (БД) провайдера информации, а также автоматическое информирование пользователей.

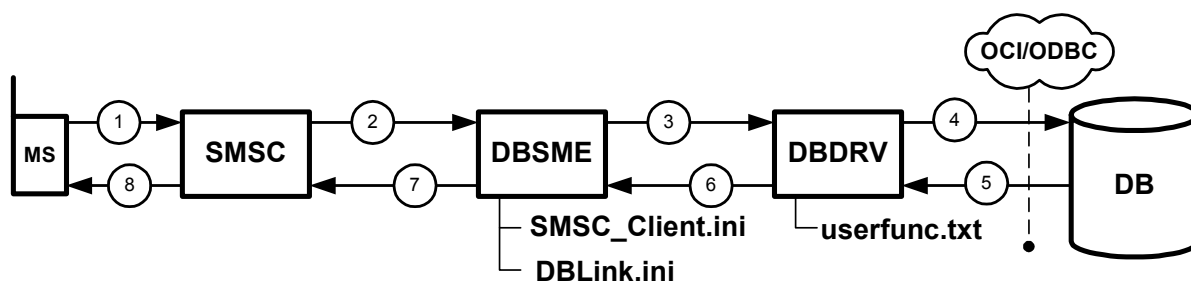
Для реализации данных функциональных возможностей к БД провайдера информации предъявляются требования, изложенные ниже.

2 Информирование по запросу

Пользователи CBOSSsms имеют возможность запрашивать информацию, хранимую в базе данных провайдера информации, посылая запросы в виде коротких сообщений.

Для описания технической стороны данного процесса воспользуемся структурной схемой, представленной на Рис. 2.1.

Рис. 2.1. Структурная схема информирования по запросу



Элементы структурной схемы:

- MS – мобильный терминал абонента сети сотовой связи.
- SMSC – сервис-центр CBOSSsms (программа сервис-центра SMSC).
- Dbsme2 – программно функциональный модуль (ПФМ), обеспечивающий интерфейс с драйверами баз данных.
- Dbdrv – драйвер базы данных, предназначенный для обеспечения взаимодействия программных модулей CBOSSsms с БД провайдера информации.

Примечание. Вместо Dbdrv может использоваться компонент интеллектуальной сети CBOSSin – cbSDP. Назначение и принцип работы обоих программно-функциональных модулей одинаков.

- SMSC_Client.ini, DBLink.ini – настроечные файлы ПФМ Dbsme2, задающие параметры для взаимодействия SMSC с Dbdrv.
- userfunc.txt – настроечный файл Dbdrv, определяющий тип и параметры транзакций к БД.

Пользователь программно-аппаратного комплекса CBOSSsms инициирует запрос к БД, в виде короткого сообщения строго определённого формата, которое он отправляет со своего мобильного терминала (точка 1 на Рис. 2.1) на заданный номер сети, например, на номер "100". **SMSC** перенаправляет данное сообщение на клиента **Dbhme2** (точка 2 на Рис. 2.1). **Dbhme2** делает синтаксический разбор параметров в соответствии с файлом **SMSC_Client.ini** (пример которого приведен ниже) и передает необходимые параметры драйверу базы данных (точка 3 на Рис. 2.1). Если параметры не удовлетворяют маске входящих запросов, то **Dbhme2** отправляет сервис центру сообщение "incorrect data", и **SMSC** информирует отправителя сообщения о некорректности отправленного сообщения.

Пример фрагмента файла **SMSC_Client.ini**, определяющего параметры запроса к базе данных:

[SMSC requests]

// Запрос к БД

number:100

input: string str

output: BillingLink, 61, "SP", _src, 2

В соответствии с представленным примером, входящими параметрами запроса являются номер получателя (number) – "100" и текст короткого сообщения (input), в данном случае, он может быть любого содержания. Исходящими параметрами, которые передаются драйверу базы данных для формирования запроса, являются:

1. Номер функции которую должен вызвать драйвер базы данных – 61.
2. Строка типов данных – "SP":
 - S – string (null terminated).
 - P – priority (unsigned char).
3. Параметры, с которыми вызывается функция:
 - _src – мобильный номер пользователя CBOSSsms, инициировавшего запрос (адрес отправителя короткого сообщения).
 - 2 – значение дополнительного параметра P (приоритет).

Примечание. В данном случае приведены не все необходимые параметры, задаваемые в настроечном файле **SMSC_Client.ini**. Полное описание формата файла дано в документе *CBOSSsms 3.0. Руководство пользователя* (см. раздел с описанием ПФМ **Dbhme2**).

Драйвер базы данных **Dbdrv** формирует на основе полученных от **Dbhme2** значений параметров SQL-запрос и выполняет его или передает полученные от **Dbhme2** параметры на вход вызываемой пакетной процедуре, хранимой в БД (точка 4 на Рис. 2.1). При этом для взаимодействия с БД могут использоваться ODBC или OCI (Oracle Call Interface).

Все возможные транзакции драйвера к базе данных определяются в настроечном файле `userfunc.txt`. Далее покажем пример фрагмента `userfunc.txt` для реализации приведённого выше запроса баланса:

- вариант для ODBC:

```
0 61 Transaction1 select поле1, поле2, ... from таблица where
полеN= •
```
- вариант для OCI (SQL-запрос):

```
0 61 Transaction2 select 'Ваш баланс равен' | |поле1 from таблица
where полеN= •
```
- вариант для OCI (вызов пакетной процедуры БД):

```
2 61 Transaction3 • O=package_1.function_1(• i,• i)
```

Примечание. В приведенном примере не представлено подробного описания структуры файла `userfunc.txt`. Описание формата файла дано в документе *CBOSSsms 3.0. Руководство пользователя* (см. раздел описани-ем ПФМ **Dbdrv**).

Результатом выполнения запроса или функции должна быть строка, включающая имя шаблона текста короткого сообщения (templates) из базы данных CBOSSsms, а также подставляемые в шаблон значения. Например, “#template=balance#amount=35”, где `balance` – имя шаблона, `amount` – имя параметра, а 35 – значение параметра (подробнее об использовании шаблонов см. документ *CBOSSsms 3.0. Руководство администратора*).

Если нет возможности предоставить результат запроса или выполнения функции в виде шаблона с перечнем необходимых для подстановки в него значений параметров, то тогда данные должны возвращаться в виде простого текста, тип – Octet String (Рис. 2.1).

Таким образом, описав соответствующий SQL-запрос или вызов хранимой в базе данных пакетной процедуры, провайдер информации может предоставить сведения из своей БД пользователю CBOSSsms.

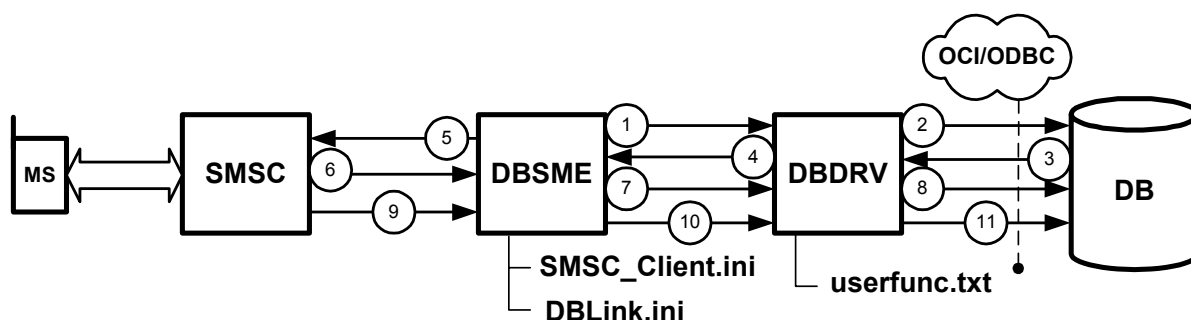
Результат запроса (текст) передаётся от **Dbdrv** к **Dbme2** (точка 6 на Рис. 2.1). **Dbme2**, получив результаты запроса, передает их **SMSC** (точка 7 на Рис. 2.1) в виде короткого сообщения для пользователя CBOSSsms, сделавшего запрос. При этом, если в тексте сообщения есть ссылка на шаблон, то **SMSC** меняет это короткое сообщение на другое, взяв за основу последнего текст шаблона на языке пользователя CBOSSsms и подставив в него указанные в исходном сообщении параметры. **SMSC**, доставит короткое сообщение с ответом на MS пользователя CBOSSsms (точка 8 на Рис. 2.1).

3 Автоматическое информирование

Провайдер информации может обеспечить регулярную автоматическую рассылку информации из своей базы данных пользователям CBOSSsms в виде коротких сообщений.

Для описания алгоритма автоматического информирования абонентов воспользуемся структурной схемой, представленной на Рис. 3.1.

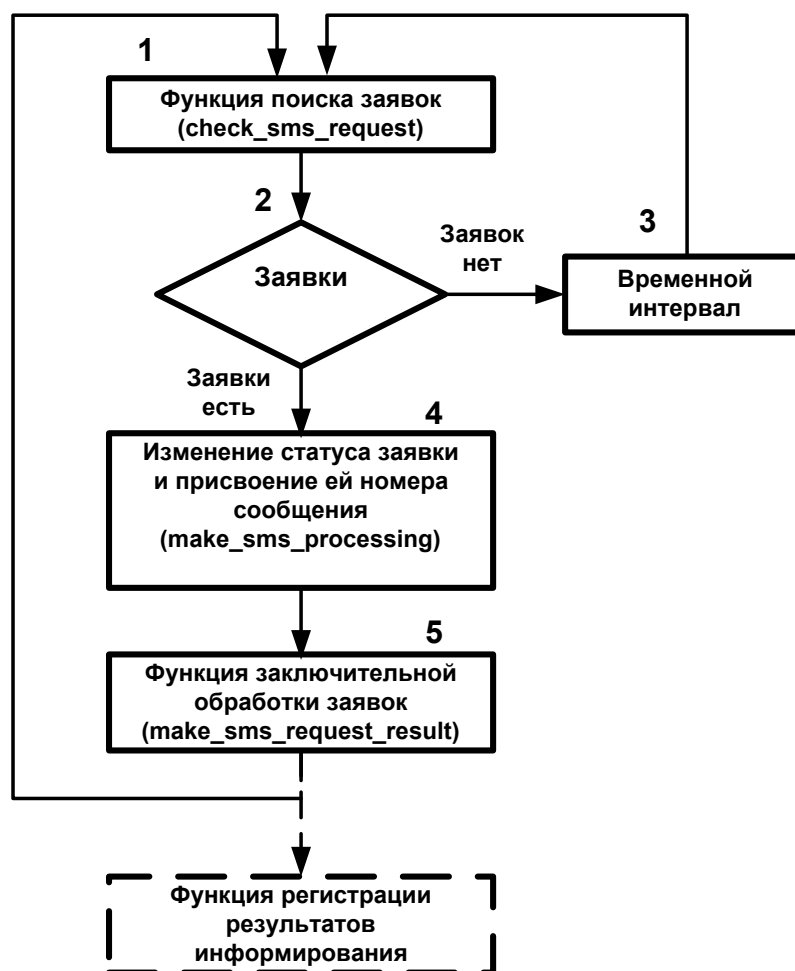
Рис. 3.1. Структурная схема автоматического информирования



Db sme2 инициирует вызов функции поиска заявок посредством драйвера базы данных **Dbdrv** (точки 1 и 2 на Рис. 3.1). Драйвер **Dbdrv** для обращения к БД использует файл **userfunc.txt**. Пример фрагмента данного файла для реализации функции **check_sms_request** и блок-схема алгоритма автоматического информирования (Рис. 3.2) приведены ниже.

Примечание. Вместо **Dbdrv** может использоваться компонент интеллектуальной сети CBOSSin – **cbSDP**. Назначение и принцип работы обоих программно-функциональных модулей одинаков.

Рис. 3.2. Блок-схема алгоритма автоматического информирования



Фрагмент файла userfunc.txt:

- вариант для ODBC:

```
0 37 Transaction1 select поле1, поле2, ... from таблица where по-
леN= •
```
- вариант для OCI (SQL-запрос):

```
0 37 Transaction2 select поле1, поле2, ... from таблица where по-
леN= •
```
- вариант для OCI (вызов пакетной процедуры БД):

```
2 37 Transaction3 • O=info_1.check_sms_request(• i,• i)
```

Примечание. В приведенном примере не представлено подробного описания структуры файла userfunc.txt. Описание формата файла дано в документе *CBOSSsms 3.0. Руководство пользователя* (см. раздел описани-ем ПФМ Dbdrv).

При отсутствии заявок функция поиска `check_sms_request` должна возвращать драйверу “-1” – нет заявок (блок 2 на Рис. 3.2).

В случае обнаружения заявки, функция должна возвращать драйверу следующие обязательные параметры:

1. Системный номер заявки.
2. Номер куда – MSISDN (C-Octet String).
3. Номер откуда – NUM_A (C-Octet String).
4. Текст сообщения или имя шаблона с подставляемыми в него значениями – DSC (C-Octet String).
5. Язык (integer):
 - 1-английский.
 - 2-русский.
6. Тип задачи (C-Octet String).
7. Приоритет: 0, 1, и т.д. (integer).
8. Начало обработки сообщения (date).
9. Дата хранения сообщения в SMSC (date).

Если результат доставки сообщения до абонента не имеет значения, то в этом случае функция должна удалить информацию о заявке из БД.

В следующий раз функция поиска заявок будет вызвана через временной интервал, настраиваемый в параметрах **Db sme2** (блок 3 на Рис. 3.2).

Данные для работы функции поиска заявок можно извлекать, например, из некоторой таблицы, например, из таблицы `INFO_LIST` со следующими полями:

Поля таблицы	Описание	Данные (пример)
N	Системный номер записи	54
TYP	Информирование абонентов	1
MSISDN	Мобильный номер	9885256
STATUS	Статус заявки	Список №1.
LAST_RESULT	Результат последней попытки	Список №2.
DSC	Простой текст или шаблон с параметрами	Баланс равен 35y.e #template=balance#amount=35
R_SMS_N	Номер ассоциированного сообщения в SMSC.	3256
NUM_A	Номер отправителя (А-номер)	0880
PRTY	Приоритет заявки	15
FD	Время начала жизни версии реализации объекта	17.07.2000
TD	Время окончания жизни версии реализации объекта	01.01.3000

В таблице должны быть созданы индексы:

- `INFO_LIST$N` – содержит поле N.
- `INFO_LIST$SMS_N` – содержит поле `R_SMS_N`.

Список (Список №1) возможных значений для поля STATUS:

- Не обработана.
- В обработке.
- Обработана.
- Снята.
- Снята по завершению времени.

Список (Список №2) возможных значений для поля LAST_RESULT:

- Короткое сообщение доставлено.
- Короткое сообщение не было доставлено.
- Нет направления для доставки короткого сообщения.
- Неуспех доставки короткого сообщения в SMSC.

Далее параметры от драйвера передаются **Db sme2** (точка 4 на Рис. 3.2), который формирует сообщение и передает его на сервис-центр SMSC (точка 5 на Рис. 3.2). **SMSC** присваивает этому сообщению определённый номер и передаёт номер **Db sme2** (точка 6 на Рис. 3.2).

Далее **Db sme2**, посредством драйвера базы данных (точки 7 и 8 на Рис. 3.2), вызывает функцию изменения статуса данной заявки на значение “В обработке” и присвоения ей номера сообщения в **SMSC** – **make_sms_processing** (блок 4 на Рис. 3.2).

Пример фрагмента файла userfunc.txt с реализацией данной функции приведен ниже.

- вариант для ODBC:

```
1 38 Transaction1 select поле1, поле2, ... from таблица where полеN= ●
```
- вариант для OCI (SQL-запрос):

```
1 38 Transaction2 select поле1, поле2, ... from таблица where полеN= ●
```
- вариант для OCI (вызов пакетной процедуры БД):

```
2 38 Transaction3 ● O=info_1.make_sms_processing(● i,● i)
```

Данная функция обновляет поле **R_MSG_N** в таблице для записи о заявке в соответствии со значением переданным от драйвера базы данных.

После обработки сообщения, **SMSC** передает **Db sme2** соответствующий код результата доставки (точка 9 на Рис. 3.1). **Db sme2** посредством драйвера базы данных **Dbdrv** вызывает функцию заключительной обработки заявок – **make_sms_request_result** (точки 10 и 11 на Рис. 3.1).

Пример фрагмента файла userfunc.txt с реализацией данной функции приведен ниже.

- вариант для ODBC:

```
1 39 Transaction1 update таблица ... set поле1... from таблица where полеN= ●
```
- вариант для OCI (SQL-запрос):

```
1 39 Transaction2 update таблица ... set поле1... where полеN= ●
```

- вариант для OCI (вызов пакетной процедуры БД):

2 39 Transaction3 • O=info_1.make_sms_request_result(• i,• I)

Данная функция (блок 5 на Рис. 3.2), в зависимости от результата доставки, должна изменить информацию о заявке:

- Если в качестве результата доставки, функции передается код 200 (короткое сообщение доставлено), то данная функция должна выставить признак обработки для данной заявки или же удалить запись о ней из базы данных (В таблице INFO_LIST обновляются поля STATUS и LAST_RESULT на значения “Обработана” и “Короткое сообщение доставлено” соответственно).
- Если результат доставки, передается в функцию как значение отличное от кода 200, например, 201 (неуспех доставки короткого сообщения), то в этом случае make_sms_request_result должна выставить признак неудачной доставки для данной заявки (В таблице INFO_LIST обновляются поля STATUS и LAST_RESULT на значения “Не обработана” и “Неуспех доставки короткого сообщения” соответственно).

Допускается регистрация информации о результатах информирования в других таблицах БД провайдера информации, с тем чтобы дать доступ к данной информации сотрудникам подразделений провайдера, но для этих целей следует использовать дополнительную функцию, как показано на Рис. 3.2.