

Приложение D – формат MAP файла

Описание базы представляет структурированный текстовый файл, состоящий из командных строк. Тип команды определяется либо идентификатором, расположенным в начале строки, либо, если таковой отсутствует, значением по умолчанию, определяющимся предыдущими командами.

Все команды начинаются символом (<) «меньше», за которым следует идентификатор длиной в 3 символа. Завершает идентификатор команды символ пробела.

Обычно команды состоят из нескольких полей. Разделителем полей является один или несколько символов пробела. Если поле содержит внутри себя пробелы, его следует заключить в кавычки. Поля можно пропускать, оставляя принятые для них значения по умолчанию, в этом случае, в качестве значения поля следует поставить символ (#). Если поле состоит из нескольких частей, или представляет собой список, разделителем частей поля является символ (|).

Иерархическая зависимость команд определяется порядком их следования - так, например, считается, что команды описания полей и свойств таблицы относятся к последней описанной таблице. Общие команды можно размещать в произвольном порядке в любом удобном месте файла описания.

Если интерпретатор файла не может распознать команду, она игнорируется. Для удобства чтения допускается вставлять в файл описания пустые строки. Команда <REM обозначает комментарий, который можно использовать для пояснения значений команд.

Заголовком файла является строка :

RGBASE V1.0

Эта строка описывает тип и версию совместимого интерпретатора для данного файла. За ней следует строка (возможно пустая), предназначенная для комментария. Интерпретатор игнорирует эту строку.

Большинство команд имеют следующую структуру:

[Имя][Заголовок][Псевдоним][Флаги][Тип][Права доступа][Формат] [Значение по умолчанию]

[Имя]	Реальное имя объекта (файл на диске, таблица в базе данных, поле в таблице).
[Заголовок]	Имя, под которым объект будет представлен пользователю программы (заголовок таблицы, название поля и т.д.)
[Псевдоним]	Сокращенное название объекта для удобства ссылки на него внутри файла описания и в шаблонах RTForms.
[Флаги]	Свойства объекта. Записывается в виде последовательности нулей и единиц (0110). 0 сбрасывает флаг, 1 - задает его.
[Тип]	Тип объекта. Применимо только для полей.
[Права доступа]	Уровни пользовательского доступа для чтения и изменения объекта. Может состоять из нескольких прав - в этом случае, они разделяются символом ' '. Если ожидается несколько прав, а указывается один уровень, этот уровень присваивается всем правам. Значение '-1' запрещает доступ независимо от пользовательского уровня. Указанные права распространяются на все подчиненные объекты.
[Формат]	Форматная строка объекта.
[Значение по умолчанию]	

Общие команды

Общие команды могут размещаться в произвольном порядке, в любом удобном месте.

<APP [Имя]

Пример

<APP "B"

Задает тип приложения для шаблонов RTForms. Базы данных должны указывать тип 'B'.

<USR [Имя] [Уровень доступа]

Пример

<USR "IVAN" 9

Задает уровень доступа для данного пользователя (имя компьютера в сети). Уровень доступа это целое неотрицательное число. Чем больше уровень доступа, тем большими правами в данной базе наделен пользователь. При работе с различными объектами, программа сравнивает уровень доступа пользователя с уровнями доступа, указанными для объекта и на основании этого регулирует возможности пользователя.

<AKA [Имя] [Псевдоним]

Пример

```
<AKA "Z" "B.O.Z"
```

Позволяет задавать сокращенные имена (псевдонимы) для часто используемых в файле описания MAP и RTForms выражений. Когда интерпретатор встречается псевдоним, он подставляет заданное выражение до начала вычисления.

<FLD [Переменная] [Значение]

Пример

```
<FLD "FIRM" "РУМБ"
```

Текстовое значение пользовательской переменной. В данной реализации поддерживается только переменная FIRM для использования индекса заданной организации в качестве параметра таблицы.

<REM [Комментарий]

Пример

```
<REM Связи полей должны идти после их описания (см. примечание к вычисляемым полям)
```

Позволяет снабжать описания команд комментариями. Интерпретатор игнорирует эти строки.

<LOG

Пример

```
<LOG
```

Обозначает окончание файла описания. Если подобная команда встречается в файле описания, на ней интерпретатор завершает обработку, все последующие строки игнорируются. В противном случае, обработка продолжается до конца файла.

Команды описания базы данных

<MDB [Имя] [Заголовок] # [Флаги] [Права доступа]

Пример

```
<MDB "Video" "База видеокассет" # 11 9
```

Задаёт параметры базы данных.

[Имя]	Имя базы данных. Перед созданием базы данных, в каталоге, где размещен файл настроек, создается подкаталог с данным именем. Этот подкаталог становится корнем всех подкаталогов базы (документы, шаблоны...) Сам файл базы данных также размещается в этом каталоге. Если этот параметр не задан, в качестве имени базы используется имя файла настроек (без расширения).
[Заголовок]	Строка, выводимая в качестве заголовка программы при открытии базы данных. Если этот параметр не задан, используется имя базы. За заголовком следует пустое поле '#' для сохранения общей структуры командной строки.
[Флаги] :	<p>a - Авторизация доступа. Если этот флаг задан, при открытии базы данных программа попросит пользователя указать логин и пароль, в противном случае, имя пользователя будет соответствовать имени его компьютера в сети. Также, при задании этого флага становится возможным сменить имя пользователя в процессе работы с базой (кнопка "Смена пользователя").</p> <p>b - Защита базы данных. При задании этого флага файл базы данных создается с уникальным паролем. Флаг необходимо задавать перед созданием базы данных и нельзя изменять.</p>

За флагами следует пустое поле '#' для сохранения общей структуры командной строки.

[Права доступа] :	a - Минимальный уровень доступа для удаления записей. Задаёт уровень удаления для всех таблиц базы. В описаниях таблиц можно задать индивидуальные уровни удаления превышающие данный.
-------------------	--

Команды описания таблицы

<TBL [Имя] [Заголовок] [Псевдоним] [Флаги] # [Права доступа]

Пример

```
<TBL "Store" "Склад кассет" "SC" 00011 # 0101910
```

[Имя]	Физическое имя таблицы в базе данных. Несколько таблиц могут иметь одно и то же физическое имя - в этом случае, все пересекающиеся поля (поля, описанные в обеих таблицах) будут отображать одни и те же данные. С точки зрения программы, таблица рассматривается скорее как представление
-------	---

данных - оно может быть отфильтровано с помощью параметров фильтра (<FLT), отображать только часть полей физической таблицы, отображать вычисляемые и взаимосвязанные поля. Псевдоним должен отличаться.

- [Заголовок] Заголовок таблицы, отображаемый программой в соответствующей закладке.
- [Псевдоним] Краткое имя, которое можно использовать, например, для ссылки на данную таблицу в файле описания.
- [Флаги]:
- a - Если флаг задан, таблица считается иерархической (см описание <LVL).
 - b - Если флаг задан, таблица считается подчиненной. Подчиненная таблица - используется для отображения подчиненных данных других таблиц, не отображается в списке таблиц базы. Одна и та же подчиненная таблица может быть использована для хранения данных из разных таблиц (например, ссылка на телефонные номера). Это возможно благодаря тому, что подчиненная таблица хранит идентификаторы как записи, так и таблицы "владельца ссылки".
 - c - Если флаг задан, таблица не будет отображаться в списке таблиц базы.
 - d - Если флаг задан, таблица считается фильтруемой. Без указания этого флага для таблицы невозможно будет вызвать диалог пользовательского фильтра.
 - e - При задании этого флага будет производиться сохранение истории изменения таблицы. Для отображения истории изменения записи следует добавить в описание таблицы подчиненную таблицу [HST], например:
07 "History" "История" # 0 TBL 9 "[HST]"

За флагами следует пустое поле '#' для сохранения общей структуры командной строки.

- [Права доступа] :
- a - Чтение.
 - b - Редактирование.
 - c - Удаление.
 - d - Добавление.

<CRD [Имя файла] [Заголовок]

Пример

```
<CRD "card.rtf" "Карточка видеокассеты"
```

Позволяет пользователю задавать RTForms шаблон для отображения записи таблицы. Отображается в виде закладки с указанным заголовком. Карточку можно выводить на принтер.

<LST [Форматная строка]

Пример

```
<LST "VALID(User, 'Клиент не указан') & ' - ' & Item.Name & ' (' & Item.[TYPE] & ') '"
```

Формат отображения записи в списке. Для перемещения по записям таблицы используется список записей, отображаемый в левом окне программы. Для формирования строки отображения можно использовать названия полей таблицы, выражения и формулы RTForms, ссылки на поля других таблиц базы. Форматная строка используется также при алфавитной группировке записей.

<SRT [Критерии сортировки]

Пример

```
<SRT "[TYPE], Name"
```

Задаёт порядок сортировки записей в таблице. Представляет собой фрагмент SQL выражения (после команды SORT BY). В данном случае использовать команды RTForms нельзя, так как строка, после замены ключевых слов непосредственно используется для составления SQL-описания таблицы.

<FLT [Критерии фильтра]

Пример

```
<FLT "Action = 2"
```

Задаёт критерии отображения записей в таблице. Представляет собой фрагмент SQL выражения (после команды WHERE). В данном случае также нельзя использовать команды RTForms, так как строка, после замены ключевых слов используется для составления SQL-описания таблицы.

<LVL [Имя] [Список полей]

Пример

```
<LVL "Данные об организации" "1|2|3|4|5|6|7|8|11|10|12|13|9|15|14|27|28|24|25"  
<LVL "Данные о частном лице" "16|17|18|20|19|7|8|21|22|23|24|25"
```

Описывает уровни иерархических таблиц. Иерархические таблицы позволяют отображать структурные деревья, например, структуры организаций с филиалами, отделами, подразделениями и сотрудниками. Организация, отдел, частное лицо и т.д. представляются, в данном случае, как уровни иерархического дерева - организации состоят из отделов, а отделы, в свою очередь, из сотрудников. Таблица объединяет записи разных уровней и поддерживает взаимные связи. Некоторые поля таблицы универсальны для всех уровней (например, адрес, телефон). Но к большинству уровней применимы уникальные атрибуты - дата окончания Вуза важна для сотрудника, а банковские атрибуты - для организации. Для того, чтобы показать программе, какие поля использовать для каждого уровня, используются списки полей, содержащие порядковые номера актуальных для уровня полей таблицы, разделенные символом '|'. Уровням присваиваются порядковые номера в порядке их следования в файле.

<DEP [Имя поля] [Формулы расчета]

Пример

```
<DEP "Price" "UnitPrice * Quantity | PriceNDS / 1.2"  
<DEP "PriceNDS" "UnitPrice * Quantity * 1.2 | Price * 1.2"  
<DEP "UnitPrice" "Price / Quantity | PriceNDS / Quantity / 1.2"
```

Некоторые поля в таблице могут быть взаимосвязаны - если задано значение одного поля, значения других полей в группе вычисляются на основании определенных выражений, и наоборот. В данном примере, взаимосвязаны три поля (в вычислениях фигурирует также Quantity - количество штук товара, но оно не зависит от других полей). Итак, если задана цена штуки товара (UnitPrice), расчет общей стоимости (Price) и стоимости с НДС (PriceNDS) вычисляется как (UnitPrice * Quantity) и (UnitPrice * Quantity * 1.2) соответственно. Если задана общая стоимость, уже цена штуки и стоимость с НДС становятся вычисляемыми. Для описания этих связей используется команда <DEP. В ней указывается имя связанного поля и перечисляются (через символ '|') RTForms формулы расчета поля, если его значение не указано явно. Ограничение реализации: связи полей должны описываться после их описания. Это ограничение касается также вычисляемых полей.

Команды описания полей таблицы

[Номер поля] [Имя] [Заголовок] [Псевдоним] [Флаги] [Тип] [Права доступа] [Формат] [Значение по умолчанию]

Пример

```
02 "Name" "Наименование товара" # 0 TXT 0|5|1 # "Изделие"
```

Поскольку описания полей являются самыми многочисленными, для них не существует специального идентификатора. Все записи без идентификатора, следующие за описанием таблицы (<TBL), рассматриваются как описания полей.

[Номер поля]	Указывается для облегчения расшифровки карт уровней (см. описание <LVL). В данной позиции должен быть указан номер (не обязательно порядковый). Интерпретатор игнорирует этот номер - он используется только для облегчения чтения файла описания. Пустое поле ('#') недопустимо (см. ниже).
[Имя]	Реальное имя поля в таблице.
[Заголовок]	Заголовок поля, предоставляемый пользователю.
[Псевдоним]	Краткое название поля для использования в RTForms и файле описания.
[Флаги] :	а - При задании этого флага, диалог пользовательского фильтра не будет показывать данное поле.
[Тип]	Тип данных поля. Поддерживаемые типы: TXT - строка до 255 символов; MEM- строка без ограничения длины. В зависимости от версии менеджера баз данных, установленного в системе, на такое поле могут налагаться определенные ограничения. Так, например, в DAO версии 3.5 невозможны сортировка и поиск по полям этого типа; INT - целочисленное значение; DBL - число с плавающей точкой; BLN - булево значение; DAT - дата; CUR - денежная величина; LST - фиксированный список; REF - ссылка на значения поля другой таблицы; FIL - ссылка на файл; CLC - вычисляемое значение; TBL - подчиненная (см. флаги описания таблиц) или справочная таблица; MTM- таблица ссылок типа Многие-Ко-Многим; Типы TBL и MTM описывают ссылочные таблицы. Независимо от порядка их указания, они

отображаются как отдельные закладки. Подчиненная таблица используется в тех случаях, когда требуется отобразить не одно, а несколько значений для поля, например, телефонных номеров обычно бывает несколько. Справочная таблица отражает выборку записей из независимой (неподчиненной) таблицы, например, перечисляет счета, выписанные клиенту (ссылка на независимую таблицу "Счета"). Изменение данных в справочной таблице невозможно.

Для иллюстрации отношения Многие-Ко-Многим лучше всего подходит библиотечный каталог - у каждого издания может быть несколько авторов, с другой стороны, у каждого автора может быть несколько публикаций.

[Права доступа] : а - Чтение.
 б - Редактирование.
 с - Однократное редактирование. Право на однократное редактирование позволяет изменять запись только единожды (до тех пор, пока не сменилась текущая запись, значение можно менять сколько угодно, но после смены записи - нет).

[Формат] Для описания формата представления целочисленных и натуральных чисел используется нотация функции printf языка C (без указания %d и %f). Для ссылок (REF) в этом поле указывается ссылочная таблица и (если применимо) через разделитель "|" параметры фильтра, например:
 "A|[LEVEL] = 0 AND Contractor = TRUE"

Для подчиненных таблиц и таблиц MTM в этом поле указывается ссылка на таблицу.

Описание справочных таблиц отличается от описания подчиненных таблиц тем, что в этом поле указывается не только ссылочная таблица, но также RTForms выражение параметров фильтра, обозначенное символом "?". Пример:

"L|?'DateEnd IS NULL AND User = ' & [ID]"

Использование в данном случае RTForms выражения позволяет реализовать параметрический запрос, отфильтровав записи по идентификатору записи текущей таблицы.

[Значение по умолчанию] Используется для задания значения по умолчанию
 Для текстовых и числовых полей возможно задание фиксированных значений (для дат, в формате ГГГГ-ММ-ДД),
 Возможно использовать ключевые слова:
 NULL
 AUTO - автонумерация для чисел и строк,
 TODAY - текущая дата (для дат).

Для ссылочных полей (REF) можно указать ключевое слово CURRENT для подстановки предварительно выбранного пользователем в таблице-источнике значения (в текущей версии не реализовано).

У описаний вычисляемых полей (CLC) в этом поле записывается RTForms формула вычисления (ограничение реализации - вычисляемое поле должно описываться после описания полей, на которые оно ссылается).

[Имя служебного поля] ##### [Значение по умолчанию]

Пример

[OWNER] ##### [FIRM]

[LEVEL] ##### 1

Описание значения по умолчанию для служебного поля. В тех случаях, когда фильтр таблицы включает служебные поля (в данном примере, "[OWNER] = [FIRM] AND [LEVEL] = 1"), вновь созданные в этой таблице записи не смогут удовлетворять критериям ее же собственного фильтра. Явно описывать служебные поля нельзя, поэтому и применяется такая усеченная форма записи.

<ТАБ [Название закладки] [Права доступа]

Пример

<ТАБ "Персональные данные" 9

Для эффективной группировки полей используются закладки. Все поля таблицы, следующие за закладкой, считаются относящимися к этой закладке. Если закладки не используются, все поля размещаются под закладкой "Данные". Единственным правом доступа, указываемым для закладки, является право на чтение.

<RPT [Номер поля]

Пример

<RPT 1

Повтор поля в закладке - позволяет отображать одни и те же поля в разных закладках.

Ссылки на служебные таблицы

[DOC]	Ссылка на таблицу документов. Для того, чтобы для записей таблицы можно было создавать документы RTForms, необходимо добавить к ее описанию строку типа: 14 "Documents" "Документы" # 0 TBL 0 "[DOC]"
[RPT]	Ссылка на таблицу отчетов. Для того, чтобы для записей таблицы можно было создавать отчеты RTForms, необходимо разрешить для нее применение фильтра и добавить к ее описанию строку типа: 25 "Reports" "Отчеты" # 0 TBL 0 "[RPT]" Отчеты отличаются от простых документов тем, что создаются на основании нескольких выбранных записей.
[HST]	Ссылка на таблицу истории (см. описание флагов таблицы, а также пример отображения удаленных записей из всех отслеживаемых таблиц в примере файла описания).
[USR]	Ссылка на таблицу, в которой перечислены все зарегистрированные пользователи данной базы. Актуально при отображении истории изменения таблиц.
[TBL]	Ссылка на таблицу, перечисляющую все таблицы базы - позволяет получить идентификатор таблицы. Может использоваться при работе с подчиненными таблицами.

Ссылки на специальные и служебные поля

[ID]	- Служебное поле. Позволяет ссылаться на уникальный идентификатор записи.
[LEVEL]	- Служебное поле для иерархических таблиц. Ссылка на уровень записи в иерархической таблице.
[OWNER]	- Служебное поле для подчиненных таблиц. Ссылка на идентификатор записи - "владельца".
[TABLE]	- Служебное поле для подчиненных таблиц. Ссылка на таблицу - "владелец".
[CODE]	- Специальное поле. Если в описании таблицы встречается поле с данным названием (может быть текстовым или цифровым), список записей таблицы будет группироваться не по алфавиту, а по цифровому значению поля [CODE].
[TYPE]	- Специальное поле. "Общепринятое" название для спискового или ссылочного поля, отображающего тип записи.
[X], [Y], [H]	- Специальные координатные поля. Если в описании таблицы встречаются поля с данным названием, диалог фильтра отображает особые строки для работы с координатами (в текущей версии не реализовано).
[FIRM]	- Специфичная пользовательская переменная для подстановки идентификатора названия организации.

Пример MAP файла

```
RGBASE V1.0
<APP "B"
<MDB "Video" "Видеотека" # 1 9
<TBL "Store" "Каталог" "ST" 00010 # 0|0|9|-1
<LST "'(' & VALID([TYPE], 'тип не задан') & ') ' & VALID(Name, '...')"
<SRT "[TYPE], Name"
01 [CODE] "Номер" "NUM" 0 TXT 0|9|5 # AUTO
02 "Name" "Наименование" # 0 TXT 0|5
03 [TYPE] "Тип" # 0 LST 0 "Кассета|MP4|DVD|PC"
05 "Category" "Категория" "CTG" 0 REF 0|5 "СТ"
06 "StoreDate" "Дата продажи" "DT" 0 DAT 0
<REM Таблица показывает все удаленные записи в отслеживаемых таблицах
<REM В качестве имени таблицы указано физическое имя таблицы истории
<TBL "m_History" "Удаленные записи" "DEL" 0000 # 9|-1|-1|-1
<LST "'[' & Field & ']': ' & Value"
<SRT "Date DESC"
<FLT "Action = 2"
01 "Date" "Дата" # 0 DAT 0
02 "User" "Автор" # 0 REF 0 "[USR]"
03 "Field" "Таблица" # 0 TXT 0
04 "Value" "Значение" # 0 TXT 0
<USR "BOSS" 9
<USR "RECEPTION" 1
<LOG
```