



ЭНЕРГЕТИКА
МИКРОЭЛЕКТРОНИКА
АВТОМАТИКА

РАСПРЕДЕЛЕННАЯ СИСТЕМА ДИСПЕТЧЕРСКОГО УПРАВЛЕНИЯ

Crystal Reports

Руководство пользователя

РАСПРЕДЕЛЕННАЯ
СИСТЕМА
ДИСПЕТЧЕРСКОГО
УПРАВЛЕНИЯ



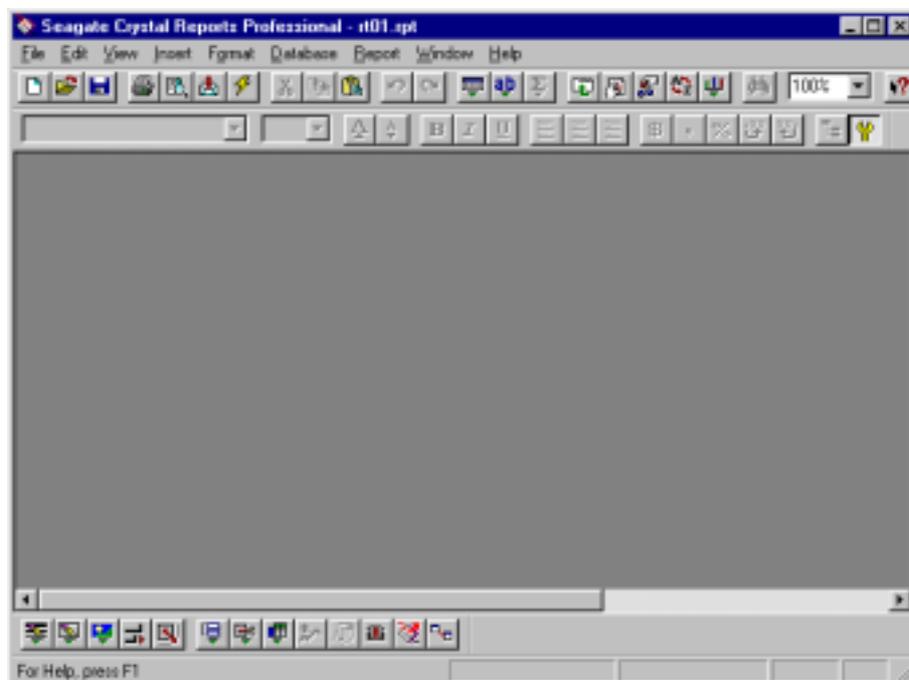
Содержание

ЗНАКОМСТВО С CRYSTAL REPORT DESIGNER	2
Окно приложения	2
Горизонтальное меню	2
Стандартная панель инструментов.....	5
Дополнительная панель инструментов.....	7
Панель форматирования	8
Строка состояния	10
Меню быстрых клавиш.....	11
Курсоры.....	13
Design Tab.....	15
Preview Tab	19
Другие базовые положения	22
ПРАКТИКА (Среда Report Design)	27
Как перемещать и изменять размеры панелей инструментов и панели форматирования	27
Как добавлять, удалять и перемещать направляющие линии	28
Как перемещать и устанавливать объекты с использованием направляющих линий	30
Как подключать/отключать сетку	32
Как увеличивать и уменьшать изображение отчета.....	32
Как отменять/восстанавливать действия.....	33
Как детализировать суммированные данные.....	34
ПРАКТИКА (Разделы и области)	35
Как добавлять, удалять, перемещать и объединять разделы.....	35
Как разбивать разделы и изменять их размеры	37
РАБОТА С БАЗАМИ ДАННЫХ	39
Обзор баз данных.....	39
Дополнительная информация	71
ПРАКТИКА (Работа с базами данных)	72
Как открывать запросы Access через DAO	72
Как открывать запросы Access через ODBC.....	73
Как открывать запросы параметров Access.....	75
Как устанавливать источник данных ODBC	78
Как проверять установки для источника данных ODBC	80
Как логироваться к источнику данных ODBC.....	81
Как добавлять таблицу базы данных ODBC в отчет	81
Как присоединиться к Серверу MS SQL через ODBC	82
Как отсоединиться от источника данных ODBC	83
Как изменить источник данных ODBC, доступный для отчета	84
Как выбирать сохраненную процедуру из базы данных SQL и изменять параметры сохраненных процедур.....	86
ИСТОЧНИКИ ДАННЫХ	89
Введение.....	89
Четыре типа данных.....	90
Файлы базы данных с прямым доступом.....	90
Источники данных ODBC	96
Файлы Crystal Query Designer	109
Словарные (Dictionary) файлы Crystal	110

ЗНАКОМСТВО С CRYSTAL REPORT DESIGNER

Окно приложения

Окно приложения просто для понимания.



- Заголовок окна идентифицирует текущий отчет и содержит стандартные кнопки Windows.
- Горизонтальное меню располагается прямо под заголовком окна.
- Стандартная панель инструментов располагается прямо под линейкой меню.
- Панель форматирования располагается под стандартной панелью инструментов.
- Дополнительная панель инструментов, если таковая установлена, располагается над строкой состояния.
- Строка состояния находится в нижней части окна.

Горизонтальное меню

Горизонтальное меню является управляющим центром для Crystal Report Designer. Каждая опция данного горизонтального меню активизирует выпадающее меню команд, которые могут использоваться для создания, модификации, печати и сохранения отчетов.

В горизонтальное меню входят следующие виды меню:

- Меню файлов (File menu)
- Меню редактирования (Edit menu)
- Меню просмотра (View menu)

- Меню вставки (Insert menu)
- Меню форматирования (Format menu)
- Меню баз данных (Database menu)
- Меню отчетов (Report menu)
- Меню окна (Window menu)
- Меню подсказки (Help menu)

Меню файлов

В меню File входят команды, которые можно использовать для открытия, закрытия и сохранения файлов, для сохранения файлов под другим именем, создания новых отчетов, почтовых меток, комбинационных таблиц, а также ряда некоторых других видов отчетов. Оно также включает в себя команду, которую можно использовать для выхода из программы, и, дополнительно, содержит команды, позволяющие просмотреть отчет перед посылкой его на печать, экспортировать отчет в различных форматах в файл на диске, послать отчет на печать, изменить поля страницы, а также выбрать нужный принтер, если Вы хотите печатать отчет не на принимаемом по умолчанию принтере. Кроме того, Вы можете добавить краткую информацию, помогающую идентифицировать отчет, и изменить установки по умолчанию таким образом, чтобы программа работала максимально эффективно. Смотрите *Команды меню файлов (File Menu Commands)* в оперативной Подсказке.

Меню редактирования

В меню Edit входят команды, которые можно использовать для модификации различных аспектов отчета. К ним относятся команды отмены и повтора действия, редактирования полей, формул, сводок, связей между подотчетами, членов OLAP, объектов OLE и связей OLE, команды просмотра примеров данных в выбранном поле, вырезания, копирования и вставки отчета и объектов OLE. Вы также можете использовать команды меню, чтобы показать, скрыть, переместить, объединить или удалить разделы отчета, удалить группы и конвертировать статические объекты OLE в редактируемые карты битов. Смотрите *Команды меню редактирования (Edit menu commands)* в оперативной Подсказке.

Меню просмотра

В меню View входят команды, которые можно использовать для модификации пользовательского интерфейса программы. Команды меню просмотра позволяют показать или скрыть стандартные и дополнительные панели инструментов, панель форматирования, строку состояния, приблизить или удалить отчет для просмотра его в различном виде, а также подключить или отключить направляющие линии, сетку и линейки. Из меню просмотра Вы можете подключать и отключать просмотр Деревя группы (Group Tree). Смотрите *Команды меню просмотра (View menu commands)* в оперативной Подсказке.

Меню вставки

В меню Insert входят команды, которые можно использовать для вставки полей базы данных, текстовых объектов, полей формул, полей параметров, объектов комбинационных таблиц, промежуточных итогов, общих итогов, сводок (чисел, средних значений и т.п.), групп, разделов и некоторых специальных полей, таких как дата печати и номер страницы. Меню вставки включает в себя также команды, позволяющие вставлять в отчет поля имен групп, подотчеты, графики, строки, окошки, диаграммы и объекты OLE. Смотрите *Команды меню вставки (Insert menu commands)* в оперативной Подсказке.

Меню форматирования

В меню Format входят команды, которые можно использовать для изменения внешнего вида элементов в отчете. К ним относятся команды для изменения шрифта и добавления границ полей, цвета фона и теней. Меню форматирования содержит команды для форматирования полей, встроенных в текстовые объекты, форматирования отдельных параграфов в этих текстовых объектах, а также целых разделов отчетов. Кроме того, данное меню содержит команды для форматирования диаграмм и осевых комбинационных таблиц. И, наконец, используя команды меню форматирования, Вы можете сделать так, чтобы программа автоматически организовывала объекты отчета, профессионально применяла указанные стили к целому отчету и перемещала объекты в стек объектов и из стека. Смотрите *Команды меню форматирования (Format menu commands)* в оперативной Подсказке.

Меню баз данных

В меню Database входят команды, которые можно использовать для добавления и удаления таблиц, используемых в отчетах, изменения псевдонима, используемого для идентификации таблицы, а также установления или разрыва связи между таблицами. Меню содержит команды для логирования и отсоединения серверов SQL и ODBC, отображения записей отчета Essbase, отображения и редактирования запросов SQL и модификации параметров сохраненных процедур. Кроме того, в меню имеются команды, заставляющие программу искать таблицы после изменения их местоположения, изменять драйверы баз данных, используемые в Вашем отчете, удалять таблицы из отчета и соответствующим образом преобразовывать отчет в случаях, когда в структуре таблиц произошли небольшие изменения. Смотрите *Команды меню баз данных (Database menu commands)* в оперативной Подсказке.

Меню отчета

В меню Report входят команды, которые можно использовать для выбора записей или групп, необходимых для включения в отчет, выбора порядка сортировки данных отчета (по записям или по группам), задания связей между подотчетами и задания даты печати отчета. Присутствуют также команды для обновления данных, использованных в отчете, и сбора всех полей, необходимых для распределения отчетов. Команды меню отчета можно использовать и для создания исполняемой версии отчета, которую совместно с Вами смогут использовать и те, кто не имеет Crystal Report Designer.

Смотрите *Команды меню отчета (Report menu commands)* в оперативной Подсказке.

Меню окна

В меню Window входят команды, которые можно использовать для изменения организации иконок и окон. В нем также присутствует список открытых окон и команда, позволяющая закрыть одновременно все окна отчета, если это необходимо. Смотрите *Команды меню окна (Window menu commands)* в оперативной Подсказке.

Меню подсказки

В меню Help входят команды, которые можно использовать для доступа к индексу оперативной Подсказки Crystal Report Designer и поиска средств, поддерживающих контекстную Подсказку. Имеются также команды для регистрации программы, доступа к технической информации о Вашей компьютерной системе и создания запроса о технической поддержке. Используя команды данного меню, Вы можете подключать и отключать диалоговое окно Welcome, а, будучи подключенным к сети Internet, сможете получить доступ к ключевым страничкам Seagate WWW. И, наконец, последняя команда предоставит информацию о версии используемой Вами программы. Смотрите *Команды меню подсказки (Help menu commands)* в оперативной Подсказке.

Стандартная панель инструментов

На стандартной панели команд Crystal Report Designer, которая постоянно остается на экране (если только Вы не отключите ее в диалоговом окне Toolbars), выведены группы часто употребляемых команд.

Стандартная панель инструментов позволяет избежать выполнения некоторых шагов, необходимых для запуска команды, и, следовательно, может в значительной степени ускорить Вашу работу по созданию отчетов. Каждая команда, доступная из панели инструментов, описана в оперативной Подсказке. (Ищите каждую команду по ее имени.)

ВНИМАНИЕ: В 32-битовой версии Crystal Report Designer Вы можете перемещать стандартную панель инструментов в другое фиксированное место в окне или устанавливать ее в виде плавающей палитры. Смотрите: Как перемещать и изменять размер панелей инструментов и панелей форматирования.

Если Вы не уверены, какую функцию выполняет кнопка на стандартной панели инструментов, установите курсор на кнопку и подержите его в этом положении. На экране появится Tool Tip (Подсказка об инструменте) с кратким описанием кнопки.

Кнопки на стандартной панели инструментов выполняют следующие функции:



Создать новый отчет



Открыть существующий отчет



Сохранить отчет



Отправить отчет на принтер



Предварительно просмотреть отчет в Preview Tab



Экспортировать отчет в файл или e-mail



Обновить данные отчета



Вырезать выделенные объекты / данные в Clipboard



Копировать выделенные объекты / данные в Clipboard



Вставить объекты / данные из Clipboard в отчет



Отменить действие



Повторить действие



Вставить поля базы данных, формулы, параметра и имени группы



Вставить текстовый объект



Вставить сумму



Активизировать Эксперта отчета



Форматировать любой раздел отчета



Установить критерий выбора записи / группы



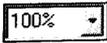
Установить порядок сортировки записи



Вставить график / схему



Найти определенную запись



Установить коэффициент увеличения для просмотра отчета



Контекстно-ориентированная Помощь

Дополнительная панель инструментов



Дополнительная панель инструментов обеспечивает быстрый доступ ко многим модернизированным экспертам (experts) и свойствам, расширяющим возможности по созданию отчетов. Активизируйте дополнительную панель инструментов нажатием кнопки Supplementary Toolbar, расположенной с правого края панели форматирования.

Кнопки на дополнительной панели инструментов выполняют следующие функции:



Вставить линию



Вставить прямоугольник



Вставить изображение



Автоматически расположить элементы отчета, чтобы они наилучшим образом располагались на странице



Активизировать Style Expert для применения профессионального стиля к отчету

-  Вставить группу
-  Вставить подотчет
-  Вставить кросс-табличный объект
-  Найти верхний и нижний номер записей или отсортировать по суммарной информации
-  Активизировать Format Editor для модификации свойств объекта
-  Компилировать отчет
-  Активизировать Report Distribution Expert
-  Определить связь между таблицами

Панель форматирования

Панель форматирования обеспечивает получение доступа ко многим популярным опциям форматирования путем нажатия кнопки или выбора из выпадающего окна. Вы просто выбираете данные, которые хотите отформатировать, а затем, для выполнения форматирования, нажимаете на соответствующую кнопку или опцию списка.

В левой части панели форматирования располагаются два выпадающих окна:



Используйте эти выпадающие окна для выбора новых шрифтов и размеров шрифтов, когда Вы не хотите использовать шрифт, принимаемый по умолчанию.

Кнопки на панели форматирования выполняют следующие функции:



Увеличить размер шрифта выделенных данных на 1 пункт при каждом нажатии кнопки



Уменьшить размер шрифта выделенных данных на 1 пункт при каждом нажатии кнопки



Изменить шрифт выделенных данных на полужирный



Изменить шрифт выделенных данных на курсив



Изменить шрифт выделенных данных на подчеркнутый



Выровнять выделенные данные по левому краю



Отцентрировать выделенные данные



Выровнять выделенные данные по правому краю



Если выделено цифровое поле, вставить денежный символ вместе с числом¹



Если выделено цифровое поле, вставить разделитель тысяч в число¹



Если выделено цифровое поле, вставить знак процента вместе с числом¹



Если выделено цифровое поле, добавить 1 знак поле точки в число¹



Если выделено цифровое поле, удалить 1 знак поле точки из числа¹



Включить/отключить Smart Navigator в групповом дереве в Preview Tab. Смотрите *Групповое дерево*



Включить/отключить дополнительную панель инструментов

¹ Программа обращается к вашей установке в разделе International Панели управления (Windows 3.x и Windows NT 3.51) или в разделе Regional Settings Панели управления (Windows 95 и Win NT 4.0).

Строка состояния

Строка состояния в нижней части окна приложения отображает ценную информацию, обеспечивающую более эффективную работу пользователей с программой.

Функции панели инструментов/панели форматирования

Если курсор находится на кнопке панели инструментов или панели форматирования, то в строке состояния отображается краткое описание функции данной кнопки.

Описания команд меню

Если Вы высвечиваете команду меню, то в строке состояния отображается краткое описание данной команды.

Текущие выборы

Если Вы высвечиваете или устанавливаете объект, в строке состояния отображается имя этого объекта (или его тип), а также информация о его местоположении и размере.

Для идентификации объекта в строке состояния отображаются:

- Слова OLE Object для рисунка или другого объекта OLE;
- Псевдоним и имя для данного поля;
- Слова Line для строки, Box для окна и Text для текстового объекта;
- Тип поля для специальных полей (Print Date, Record Number и т.д.);
- Имя для сводки или промежуточной суммы;
- Имя для формулы;
- Имя для поля параметра.

В строке состояния также отображаются:

- На каком расстоянии от левой и верхней границ раздела находится объект;
- Каков размер объекта.

Используя перечисленную информацию, Вы можете легко переместить объект или изменить его размер с высокой точностью.



Информация о поиске данных

Если Вы просматриваете свой отчет в первый раз или обновляете дату в Preview Tab, то в строке состояния отображаются четыре числа, описывающие процесс поиска данных.

Первое число определяет количество записей, выбранных из общего количества прочтенных записей.

Второе число определяет количество прочтенных записей.

Если в Вашем отчете присутствует формула выбора, то число, отражающее количество выбранных записей, будет меньше числа, отражающего количество прочтенных записей.

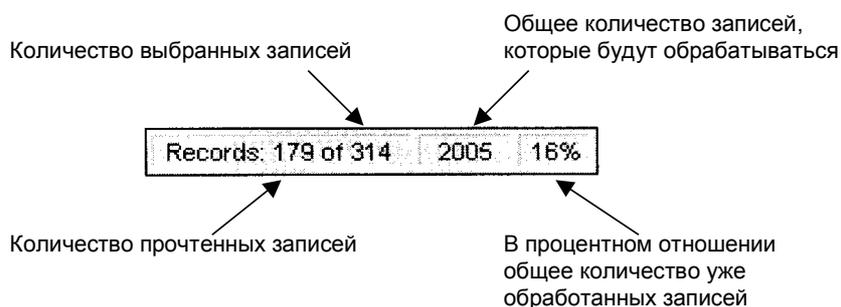
Если формула выбора в отчете отсутствует, то эти два числа всегда будут совпадать.

Третье число отражает общее количество записей, которые будут обработаны.

Если Ваш отчет основан на одной таблице, то отображенное число должно оставаться неизменным.

Если Ваш отчет основан на связанных таблицах, то, при наличии ситуаций один-ко-многим, данное число обычно увеличивается по мере того, как программа идентифицирует все связанные записи.

Последнее число отражает в процентном отношении общее количество обработанных записей.



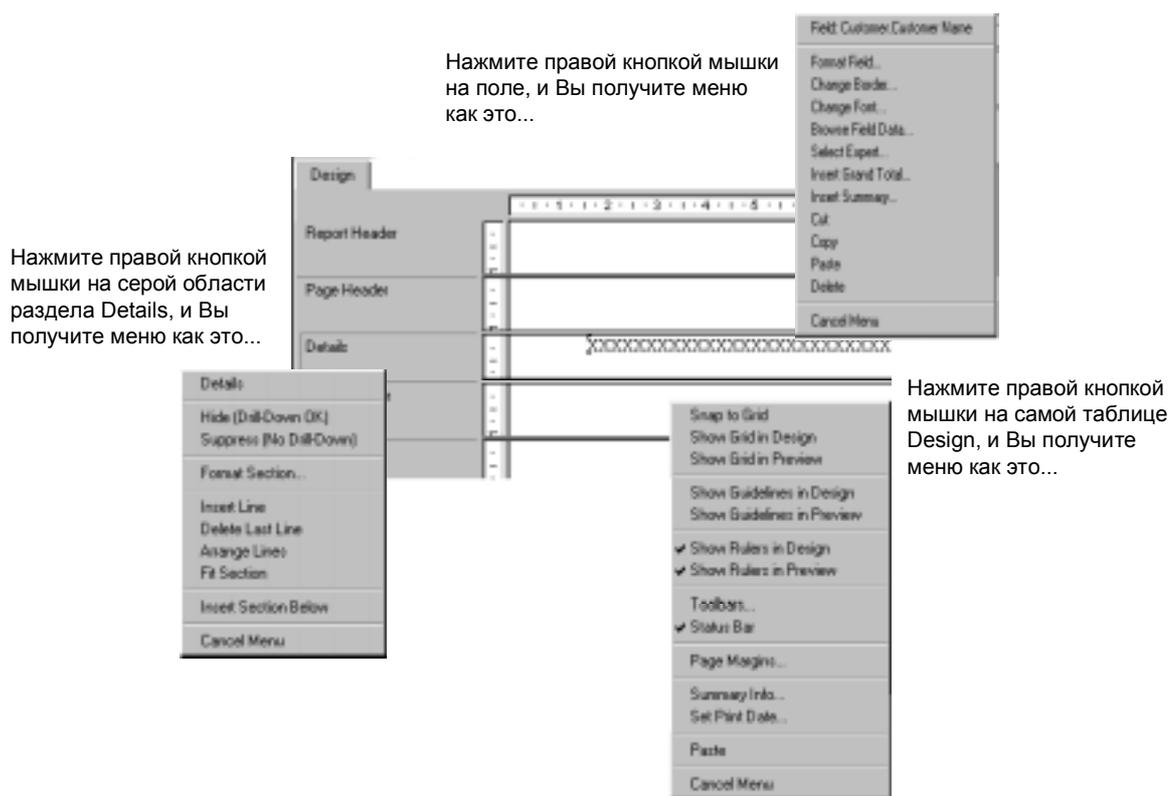
После завершения обработки отчета программа отобразит только количество выбранных и процент обработанных записей.

Меню быстрых клавиш

При работе в Design или Preview Tab Вы можете в значительной степени ускорить свою работу, если будете использовать меню быстрых клавиш. Если Вы щелкните правой кнопкой мышки по элементу отчета (рисунку, разделу, полю и т.д.), то рядом с данным элементом программа отобразит меню быстрых клавиш. В отличие от стандартных меню программы, группирующих команды по функциям (редактирование, вставка и т.д.), данное меню организовано по элементам; оно содержит только те команды, которые доступны для использования применительно к выбранному элементу.

Меню быстрых клавиш имеют большую ценность потому, что они:

- отображают имя и источник (псевдоним) элемента в верхней части меню, так чтобы Вы могли идентифицировать элементы своего отчета одним нажатием клавиши;
- упрощают изучение программы, т.к. они исключают необходимость запоминать, где следует искать команду;
- повышают эффективность работы с программой, т.к. Вы работаете только с компактным списком команд, что упрощает поиск нужной команды;
- показывают, какие операции Вы можете производить с данным элементом.



Курсоры

Crystal Report Designer в различные моменты работы использует различные курсоры:



Данный курсор является первичным. Он используется для выбора меню, выбора опций из диалоговых окон, работы с линейками прокрутки, нажатия кнопок и объектов и т.д.



Данный курсор появляется при обработке программой команды. При наличии на экране «Песочных часов» Вы не можете выбирать какую-либо команду или продолжать работать над отчетом.



Данный курсор изменяет один из ряда различных курсоров с двойной стрелкой, когда он оказывается на ручке изменения размера на графике с побитовым изображением, графическом окне, графической строке или поле.



Данный курсор появляется, когда Вы перемещаете один объект в другое место Вашего отчета.



Данный курсор появляется, когда Вы «перетаскиваете» один элемент по области, куда он может быть установлен.



Данный курсор появляется, когда Вы «перетаскиваете» несколько элементов по области, куда они могут быть установлены. Он также используется для одновременного копирования нескольких элементов.



Данный курсор появляется, когда Вы «перетаскиваете» один элемент по области, куда он не может быть установлен. Например, курсор появляется, когда Вы «перетаскиваете объект» комбинационной таблицы в раздел, в котором он не может быть установлен.



Данный курсор появляется при копировании одного элемента.



Данный курсор появляется вместо курсора Arrow, когда он устанавливается на границе любого раздела отчета. С его помощью Вы можете передвинуть линию границы раздела с целью увеличения или уменьшения размера раздела.



Данный курсор появляется, когда Вы устанавливаете указатель на левую границу раздела отчета. Если Вы нажмете при этом на кнопку мышки, то появится горизонтальная линия, которую Вы затем можете установить в том месте, в котором хотите разбить раздел.



Данный курсор появляется, когда Вы вручную устанавливаете связи в Visual Linking Expert. Смотрите *Visual Linking Topics Index* в оперативной Подсказке Crystal Report Designer.



Данный курсор появляется, когда указатель установлен на суммарное значение или график в Preview Tab. Если в это время Вы дважды щелкните кнопкой мышки по элементу графика или суммарного значения, то за ними программа отобразит детали.



Данный курсор является курсором рисования. Он появляется, когда Вы вставляете окна и линии. Стержнем «Карандаша» отмечается место, с которого начнется рисование; он также используется для определения размера и формы нарисованного объекта.



Данный курсор появляется при нажатии кнопки Help на Стандартной панели управления. Используйте его для доступа к системе оперативной Подсказки. Просто нажмите на элемент отчета, диалоговое окно и т.д., и на экране появится контекстная подсказка для данного элемента.



Данный курсор появляется в системе оперативной Подсказки. Он заменяет собой курсор Arrow, когда Вы устанавливаете его на тексте или графике, вызывающем переход к другой теме в оперативной Подсказке.



Данный курсор появляется как свойство Intellimouse для прокрутки Вашего отчета в любом направлении в Preview Tab.



Данный курсор появляется как свойство Intellimouse для прокрутки Вашего отчета вверх/вниз в Preview Tab.



Данный курсор появляется как свойство Intellimouse для прокрутки Вашего отчета вправо/влево в Preview Tab.



Данный курсор появляется как свойство Intellimouse для панорамирования Вашего отчета вверх.



Данный курсор появляется как свойство Intellimouse для панорамирования Вашего отчета вниз.



Данный курсор появляется как свойство Intellimouse для панорамирования Вашего отчета вправо.



Данный курсор появляется как свойство Intellimouse для панорамирования Вашего отчета влево.



Данный курсор появляется как свойство Intellimouse для панорамирования Вашего отчета вверх-и-вправо.



Данный курсор появляется как свойство Intellimouse для панорамирования Вашего отчета вверх-и-влево.



Данный курсор появляется как свойство Intellimouse для панорамирования Вашего отчета вниз-и-вправо.



Данный курсор появляется как свойство Intellimouse для панорамирования Вашего отчета вниз-и-влево.

Design Tab

При работе с Crystal Design Report часть Design Tab используется значительно чаще, чем любая другая.



В Design Tab выполняется большая часть первоначальной работы по созданию отчета. Она ставит указатели и метки на различные разделы отчета. Вы можете устанавливать объекты в требуемые разделы, задавать требования к сортировке, группированию и контрольному суммированию, осуществлять первичное форматирование и т.д. Смотрите *Характеристики печати областей*.

Design Tab поддерживает наиболее эффективную среду для разработки Вашего отчета, т.к. Вы работаете в таблице с представлениями данных, а не самими данными. Если Вы устанавливаете в отчете поле, то программа использует рамку для его идентификации в Design Tab; она не осуществляет поиск данных. Таким образом, Вы можете добавлять и удалять поля и другие объекты, перемещать их, устанавливать комплексные формулы и т.д., не привлекая при этом компьютерные или сетевые ресурсы, необходимые для сбора данных.

Отчет, создаваемый Вами в Design Tab, представляет собой виртуальный отчет; он содержит структуру и инструкции по созданию окончательного отчета, но сам по себе отчетом не является. Для превращения отчета в Design Tab в окончательный отчет или в отчет, для которого Вы сможете провести “тонкую отладку”, Вы просто добавляете данные. Вы можете делать это во время предварительного просмотра отчета, его печати или какого-либо другого вывода. Теперь в отчете уже будут присутствовать фактические данные.

Области Design Tab

Когда Вы начинаете создавать отчет, Crystal Report Designer автоматически создает в Design Tab пять областей.

1. Report Header (Заголовок отчета)

Данный раздел используется, в основном, для заголовка отчета и другой информации, которая, по Вашему мнению, должна присутствовать в начале отчета. Кроме того, в Заголовок отчета могут быть включены графики и комбинационные таблицы с данными, относящимися ко всему отчету.

2. Page header (Заголовок таблицы)

Данный раздел, в основном, несет информацию, которая, по Вашему мнению, должна присутствовать в верхней части каждой страницы. К ней можно отнести такие вещи, как названия глав, название документа и другую аналогичную информацию. Вы также можете использовать данный раздел для отображения заголовков полей над соответствующими полями своего отчета.

3. Details (Детали)

Данный раздел включает в себе “тело” отчета. В нем обычно представлены данные отчета. Этот раздел будет печататься один раз для каждой записи.

4. Report Footer (Нижний колонтитул отчета)

В данном разделе заключена информация, которая должна присутствовать в конце отчета, например, итоговые суммы, а также графики и комбинационные таблицы с данными, относящимися ко всему отчету.

5. Page Footer (Нижний колонтитул страницы)

Данный раздел обычно содержит номер страницы и любую другую информацию, которая, по Вашему мнению, должна присутствовать в конце каждой страницы.

Если Вы добавляете в отчет группу, сводку или промежуточную сумму, то программа создает два дополнительных раздела:

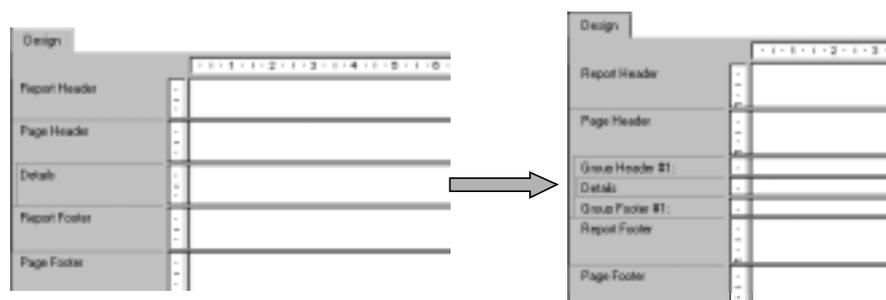
1. Group Header (Верхний колонтитул группы)

Данный раздел содержит поле имени группы и может использоваться для отображения графиков или комбинационных таблиц с данными, относящимися только к данной группе. Он печатается один раз в начале группы.

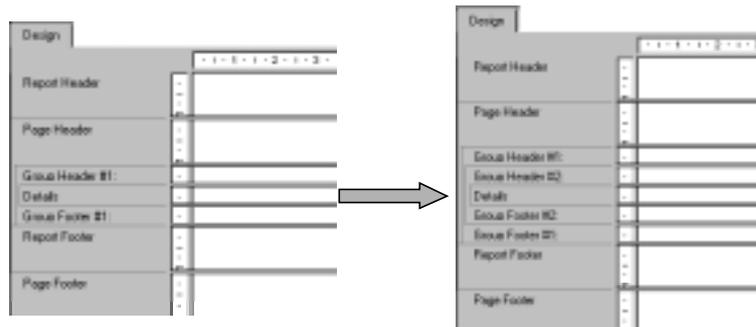
2. Group Footer (Нижний колонтитул группы)

Данный раздел обычно содержит суммарное значение, если таковое имеется, и может использоваться для отображения графиков или комбинационных таблиц. Он печатается один раз в конце группы.

Если Вы добавляете в отчет группу, сводку или промежуточную сумму, то область Group Header появится непосредственно над областью Details, а область Group Footer – сразу же под областью Details.



Если Вы будете устанавливать дополнительные группы, то программа создаст новые области групп между областью Details и имеющимися областями Group Header и Group Footer.



Подобно исходным областям, эти вновь созданные области могут содержать один или несколько разделов. По умолчанию, каждая из них будет иметь один раздел.

Характеристики печати областей

Каждая область отчета имеет свои собственные характеристики печати. Важно понимать эти характеристики, т.к. они влияют на то, *когда* и *как часто* будут печататься различные объекты отчета.

Когда печатаются области

Области печатаются в порядке их появления в Design Tab (сверху вниз). Если в области присутствует более одного раздела, то все разделы будут печататься в порядке их появления в области. Следовательно, если у Вас три раздела Report Header, то все они будут напечатаны, по порядку, перед печатью разделов в области Page Header.

Как часто печатаются объекты

Принятие решения о том, в какое место в Design Tab следует установить объекты, упростится, если Вы поймете, как часто печатается каждая область. Особенно полезно данная информация становится, когда возникает необходимость решать, куда следует устанавливать графики, комбинационные таблицы и формулы, чтобы стало возможным достичь определенных результатов.

Объекты печатаются следующим образом:

- Объекты, размещенные в области Report Header, печатаются один раз в начале отчета.
 - Графики и комбинационные таблицы из данной области содержат данные, относящиеся ко всему отчету.
 - Формулы из данной области вычисляются один раз в начале отчета.
- Объекты, размещенные в области Page Header, печатаются в начале каждой страницы.
 - В данный раздел нельзя устанавливать графики или комбинационные таблицы.
 - Формулы из данной области вычисляются по одному разу для каждой станицы, и результаты печатаются в начале каждой новой страницы.

- Объекты, размещенные в области Group Header, печатаются в начале каждой группы.
 - Графики и комбинационные таблицы, установленные в данной области, содержат данные, относящиеся только к группе.
 - Формулы из данной области вычисляются по одному разу для каждой группы, и результаты печатаются в начале каждой группы.
- Объекты, размещенные в области Details, печатаются для каждой новой записи.
 - В данный раздел нельзя устанавливать графики или комбинационные таблицы.
 - Формулы из данной области вычисляются по одному разу для каждой области, и результаты печатаются в начале каждой записи.
- Объекты, размещенные в области Group Footer, печатаются в конце каждой группы.
 - Графики и комбинационные таблицы, установленные в данной области, содержат данные, относящиеся только к группе.
 - Формулы из данной области вычисляются по одному разу для каждой группы, и результаты печатаются в конце каждой группы.
- Объекты, размещенные в области Report Footer, печатаются один раз в конце отчета.
 - Графики и комбинационные таблицы из данной области содержат данные, относящиеся ко всему отчету.
 - Формулы из данной области вычисляются один раз в конце отчета.
- Объекты, размещенные в области Page Footer, печатаются в конце каждой страницы.
 - В данный раздел нельзя устанавливать графики или комбинационные таблицы.
 - Формулы из данной области вычисляются по одному разу для каждой станицы, и результаты печатаются в конце каждой новой страницы.

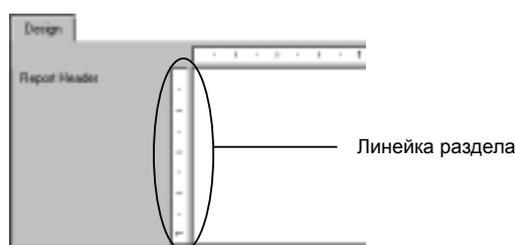
Идентификация областей и разделов и работа с ними

По умолчанию каждая область содержит только один раздел. Его имя отображается слева от раздела. Если у Вас в области размещено несколько разделов, то эти разделы будут отмечены как **a**, **b**, **c** и т.д.

ВНИМАНИЕ: Если Вы подключили окно флажка *Show Short Section Names in Design* в диалоговом окне *File Options*, то для идентификации каждого раздела будут использоваться такие сокращения, как *RH*, *PH*, *D*, *PF*, *RF*. Смотрите *File Options dialog box* в оперативной Подсказке *Crystal Report Designer*.

ВНИМАНИЕ: Если Вы щелкните правой кнопкой мышки по затененной области, содержащей имя раздела, то на экране отобразится меню быстрых клавиш с опциями, относящимися к этому разделу. Если Вы щелкните правой кнопкой мышки по затененной области слева от имен разделов, то на экране отобразится меню быстрых клавиш с опциями, относящимися к этой области.

Непосредственно слева от каждого раздела программа отображает линейку для данного раздела. Линейка раздела используется для добавления, удаления и перемещения направляющих линий, а также для визуальной ссылки при установке объектов. Смотрите: *Как добавлять, удалять и перемещать направляющие линии.*



При добавлении Вами нового раздела программа создает для него линейку. Смотрите: *Как добавлять, удалять, перемещать и объединять разделы.*

Другие возможности Design Tab

В Design Tab встроены также и некоторые другие возможности.

- Вы можете изменить размер раздела, передвигая его границы. Смотрите: *Как разбить и изменить размеры разделов.*
- Вы можете разбить раздел (создать из одного раздела два), щелкнув мышкой по его левой границе. Смотрите: *Как разбить и изменить размеры разделов.*
- Вы можете добавить горизонтальные и вертикальные направляющие линии, щелкнув мышкой по линейкам. Смотрите: *Как добавлять, удалять и перемещать направляющие линии.*
- Вы можете увеличить и уменьшить изображение отчета в диапазоне от 25% до 400% от его оригинального размера. Смотрите: *Как уменьшать и увеличивать изображение отчета.*

Preview Tab

Если перед печатью отчета Вы хотите его предварительно просмотреть, щелкните мышкой по кнопке **Print Preview** на стандартной панели инструментов.

Программа соберет данные, произведет необходимые вычисления и отобразит отчет в электронном виде в Preview Tab. После этого, Вы можете проверить размещение и форматирование своего отчета и посмотреть результаты сводок, вычислений формул, а также выбранные записи и группы.

В соответствии с принципом WYSIWYG (Видишь то, что имеешь), Вы можете непосредственно работать с этими реальными данными, проводить окончательную доводку отчета до тех пор, пока он не будет выглядеть так, как Вам требуется.

Программа работает с данными следующим образом:

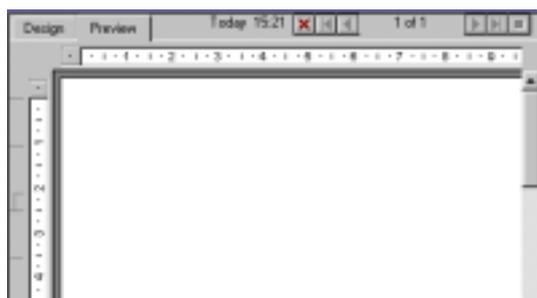
- При первом использовании Preview Tab она осуществляет поиск данных в базовых источниках данных и сохраняет их в отчете (если только Вы не ввели установку не сохранять данные).
- Затем, когда Вы предварительно просматриваете отчет, программа использует сохраненные данные, если только Вы специально не обновили данные или не добавили поля, потребовавшие от программы поиска новых данных.

Crystal Report Designer поддерживает два вида предварительного просмотра отчета:

1. В стандартном виде.
2. В виде Группового дерева.

Стандартный вид

В стандартном виде отчет отображается постранично. Используя навигационные кнопки в Preview Tab, Вы можете перемещаться в любом направлении от начала до конца отчета или же просматривать отчет постранично в обоих направлениях. Для более коротких отчетов или в случаях, когда Вас интересуют только суммы, представленные в нижних строках, Стандартный вид также предоставляет все необходимые функциональные возможности.



Индикатор Data Age (Возраст данных)

Индикатор Data Age показывает дату последнего обновления или нахождения данных, в зависимости от того, что произошло позже. Если данные были найдены или обновлены сегодня, то индикатор показывает время, когда это произошло. Смотрите *Save Data with Report command* в оперативной Подсказке Crystal Report Designer.

Групповое дерево



Вы можете активизировать и отключать вид группового дерева с помощью кнопки **Toggle Group Tree** на панели форматирования.

Вид группового дерева представляет разделенный экран.

- На правой панели экрана отображен отчет.
- На левой панели отображена схема отчета, показывающая иерархию групп и подгрупп в обычном формате дерева.

Если Вы нажмете на узел дерева, представляющий группу, программа немедленно перейдет к той части отчета, в которой содержится информация о данной группе. Для более длинных отчетов или в случаях, когда Вам необходимо постоянно перемещаться в различные точки отчета, Вы можете использовать возможности Smart Navigation.

Сходства и отличия от Design Tab

Preview Tab поддерживает те же возможности форматирования, что и Design Tab. Меню (как горизонтальное меню, так и меню горячих клавиш), стандартные и дополнительные панели инструментов, линейки и панель форматирования в Preview Tab остаются активными, обеспечивая те же функциональные возможности, что и при работе над отчетом в Design Tab. Однако, если Вам необходимо внести большое количество изменений, это лучше это сделать в Design Tab.

- Существует внутренняя связь между Design Tab и Preview Tab. Любые изменения, производимые в Design Tab, отражаются и в Preview Tab, и наоборот.
- В отличие от Design Tab, имеющей линейки для каждого раздела, Preview Tab имеет одну вертикальную линейку слева от таблицы. Функциональные возможности этих линеек одинаковы.
- Preview Tab идентифицирует разделы отчета в затененной области слева от данных. Вы можете мгновенно определить, из какого раздела отчета печатаются данные. Если в Design Tab имена разделов появляются только единожды, в Preview Tab они печатаются каждый раз, когда печатается раздел.
- В Preview Tab также активны: счетчик Записей (см. ИНФОРМАЦИЯ О ПОИСКЕ ДАННЫХ), индикатор Data Age (См. ИНДИКАТОР DATA AGE) и элементы управления Page Forward/Page Back (см. Preview Tab).
- При выборе Вами поля Preview Tab высвечивается каждое значение, а в Design Tab высвечивается только рамка поля.

Различное восприятие

Работая в Preview Tab и в Design Tab, Вы получаете различные ощущения.

Каждое поле в базе данных содержит десятки, сотни и даже тысячи значений, в зависимости от количества имеющихся в базе данных записей. Когда Вы устанавливаете поле в Design Tab, все эти значения представлены в рамке одного поля. Если Вы высвечиваете поле, на рамке появляются ручки изменения размера, и рамка изменяет свой цвет.

В отличие от этого, в Preview Tab Вы работаете с фактическими данными. Вместо рамки поля, представляющей значения различных полей, здесь появляются сами значения.

- Если Вы высвечиваете значение поля или поля формулы, то Вы фактически выбираете каждое значение в поле.
 - Программа устанавливает рамку размера вокруг выбранного Вами значения.
 - Она высвечивает все остальные значения из данного поля.
- Аналогичным образом, если Вы выбираете суммарное значение, то Вы фактически выбираете все соответствующие суммарные значения.
 - Программа устанавливает рамку размера вокруг выбранного Вами значения.
 - Она высвечивает все соответствующие суммарные значения.

За исключением явных различий во внешнем виде, процесс построения и модификации отчета одинаков в Design Tab и Preview Tab. В обоих случаях Вы почувствуете, как легко Вам будет работать с отчетами.

Другие базовые положения

Среда отчетов Crystal Report Designer отличается большой гибкостью.

- Вы можете подключать фиксацию сетки, устанавливать сетку на максимум до 1" и делать ее видимой или невидимой в Design Tab, Preview Tab или обоих режимах (см. Как подключать и отключать сетку).
- Вы также можете работать без сетки, устанавливая объекты в любое место отчета (см. Свободная форма и Свободная форма с направляющими линиями).
- И, наконец, Вы можете использовать направляющие линии, если хотите выровнять и изменить размеры объектов с высокой точностью (см. Как перемещать и устанавливать объекты, используя направляющие линии).

Устанавливайте среду так, чтобы она наилучшим образом отвечала Вашим потребностям.

Сетка

Сетка представляет собой набор координат рядов и колонок. Если сетка активна, то программа позволяет устанавливать объекты только в этих точках координат, но не между ними. Активная сетка в значительной степени упрощает установку и размещение в отчете данных, а также выравнивание объектов, если это необходимо. Если Вы попытаетесь установить объект между точками координат, программа сразу же "привяжет" этот объект к сетке, т.е. автоматически переместит объект к ближайшему пересечению координат рядов/колонок.

Размер сетки и то, будет ли она видимой, задается посредством Layout Tab в диалоговом окне File Options. По умолчанию сетка неактивна. Смотрите: *Как подключать и отключать сетку.*

Свободная форма

В отличие от более ранних версий Crystal Report Designer, в данной версии Вы можете работать без сетки, в среде свободной формы, аналогичной среде программы рисования. Свободная форма означает, что Вы можете устанавливать объекты в любом месте отчета.

Единственным ограничением является то, что программа запрещает устанавливать объекты графиков и комбинационных таблиц в разделы Page Header, Page Footer и Details. Смотрите: *Характеристики печати областей.*

Для работы в среде свободной формы, отключите флажок Snap To Grid с помощью Layout Tab в диалоговом окне File Options. Для получения дополнительной информации смотрите *диалоговое окно File Options* в оперативной Подсказке Crystal Report Designer.

Свободная форма с направляющими линиями

Вы также можете работать в среде свободной формы и иметь при этом возможность выравнивать объекты или изменять их размер как группы. Это возможно при использовании направляющих линий.

Направляющие линии представляют собой линии, идущие вертикально и горизонтально от линеек Design и Preview Tab. Эти линии обладают свойством привязки; если Вы перемещаете объект в пределах активности направляющих линий, то происходит автоматическая привязка объекта к такой линии.

- После того, как объект оказывается привязанными к направляющей линии, любое перемещение линии вызывает одновременное соответствующее перемещение с ней объекта.
- Если к направляющей линии привязано несколько объектов, то, при перемещении линии, они все будут перемещаться вместе с ней.
- Если несколько объектов привязаны к направляющей линии с двух сторон (справа и слева или сверху и снизу), то, при перемещении одной из линий, в одинаковой степени изменятся размеры всех объектов.

Использование направляющих линий в среде свободной формы обеспечит Вам гибкость управления. Смотрите: *Как добавлять, удалять и перемещать направляющие линии* и *Как перемещать и устанавливать объекты с использованием направляющих линий.*

Разделы и объекты

Crystal Report Designer позволяет вставлять в отчет различные объекты.

- **Объекты полей**

Поля из таблиц базы данных и из результатов вычислений формул, параметры, имя группы, запросы и сохраненные процедуры. Смотрите *How to insert database fields*.

- **Текстовые объекты**

Символы, слова и даже целые документы. Смотрите *How to insert text objects*.

- **Рисованные объекты**

Карты битов - *.bmp, *.pcx, *.tif, *.tga, *.jpeg. Смотрите *How to insert a picture*.

- **Графические/схематические объекты**

Графики, отображающие суммированные данные. Смотрите *Graphing*.

- **Объекты подотчета**

Отчеты, входящие в состав других отчетов, автономные или связанные с данными первичного отчета. Смотрите *How to insert a subreport*.

- **Объекты комбинационных таблиц**

Отчеты в виде электронных таблиц, позволяющие определять тенденции. Смотрите *Cross-tab objects*.

- **Объекты OLE**

Рисунки, электронные таблицы, текст и другие объекты, созданные в приложениях сервера OLE. Смотрите *OLE Objects Overview*.

Объекты можно сравнить с контейнерами. Они могут содержать данные и, в некоторых случаях, другие объекты (например, текстовый объект может содержать объекты полей, равно как и текст, а метки в объекте комбинационная таблица, фактически, являются текстовыми объектами). Каждый объект обладает свойствами, определяющими поведение данного объекта в отчете.

Вы можете устанавливать для объектов свойства атрибутов, условные свойства или и те, и другие вместе.

- Фиксированные свойства устанавливаются с использованием опций диалогового окна.
- Условные свойства устанавливаются с использованием специальных формул.

Смотрите: *Conditional Formatting* и *Absolute Formatting*.

Вы можете вставлять большинство объектов в большинство разделов отчета; однако программа накладывает некоторые ограничения на установку определенных объектов в определенные разделы из-за бессмысленности подобного действия. Например, т.к. раздел Details печатается для каждой записи, установка объекта комбинационной таблицы в раздел Details привела бы к созданию отчета

комбинационной таблицы для каждой записи, что не принесло бы никакой пользы. В связи с этим, программа запрещает установку комбинационной таблицы в раздел Details. Смотрите: краткое описание ограничений, накладываемых на разделы/объекты в разделе *Характеристики печати областей*.

ВНИМАНИЕ: Смотрите информацию о печати объектов в разделах, в которые они не могут быть физически установлены, в разделе *How to make an object underlay a following section(s)*.

Вас не должна беспокоить проблема установки объекта в такие места, куда он не подходит; программа сама позаботится об этом. Для объектов, разрешенных программой для установки в данном разделе, помните, что разрешение на установку еще не означает необходимость присутствия данного объекта в определенном разделе. Это совсем другая ситуация, и она требует принятия решения именно от Вас. Например, если Вы устанавливаете рисованный объект в:

- раздел Report Header, то он будет печататься один раз в начале отчета.
- в раздел Group Header, то он будет печататься по одному разу для каждой группы.
- в раздел Details, то он будет печататься по одному разу для каждой записи.

В такой ситуации решение о том, куда Вы установите объект, зависит от того, какого результата Вы хотите достичь. Решение остается за Вами.

Для разделов, как и для объектов, Вы также можете устанавливать фиксированные и условные свойства. Смотрите: *Conditional Formatting*.

Принадлежность объектов

По умолчанию, если Вы устанавливаете объект в раздел:

- раздел расширяется, чтобы включить в себя этот объект, если это необходимо;
- при печати объект печатается в том разделе, в который он установлен.

Однако, если Вы устанавливаете объект в раздел, которому, в соответствии с вашей установкой, принадлежат последующие разделы, то:

- при печати данного раздела объект также будет в нем печататься, но
- он будет принадлежать и последующим разделам.

ВНИМАНИЕ: Объекты, установленные в раздел, могут принадлежать всем разделам вплоть до (но не включая) "сестринского" раздела. Например, разделу Page Header могут принадлежать все разделы вплоть до (но не включая) раздела Page Footer.

Это позволяет Вам добиваться некоторых интересных эффектов в отчете. Например, Вы можете:

- печатать объект таким образом, чтобы он появлялся один раз в разделе Details параллельно с другими деталями (например, график сравнения показателей продаж параллельно с информацией для регионов);
- печатать водяной знак марки компании в центре каждой страницы;
- использовать сканированную карту битов формы в качестве руководства при посылке отчета на печать на заранее подготовленных печатных формах.

Используя возможность Underlay, Вы можете добиться потрясающих визуальных эффектов. Для большей информации смотрите *How to make an object underlay a following section(s)*.

ПРАКТИКА (Среда Report Design)

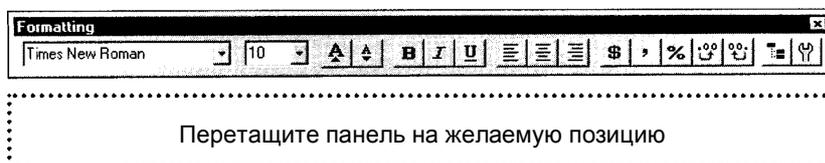
Как перемещать и изменять размеры панелей инструментов и панели форматирования

32-битовая версия Crystal Report Designer имеет перемещаемые, способные изменять размер стандартную и дополнительную панели инструментов и панель форматирования.

Вы можете перемещать эти панели в какое-то фиксированное положение или превращать их в плавающие палитры. Перемещение и изменение размеров этих панелей осуществляется следующим образом:

Перемещение

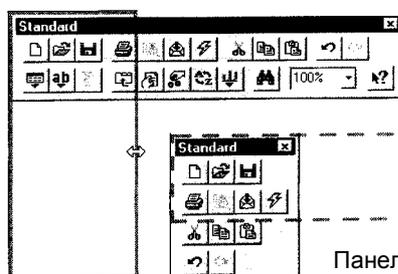
Для перемещения панели щелкните мышкой по тому месту панели, где нет кнопок и, не отпуская кнопки мышки, "перетащите" ее на новое место.



Изменение размера

Для изменения размера панели установите указатель на угол панели. После появления указателя изменения размера, двигая его внутрь или наружу, Вы можете изменить размер панели.

Перейти от этой...

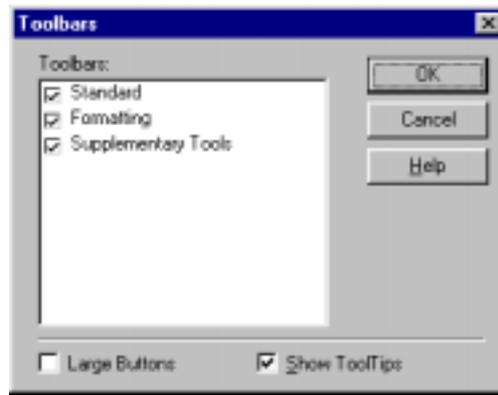


Перетащите курсор для создания желаемого размера

к этой...

Панель изменится так

С помощью диалогового окна Toolbars Вы можете подключать или отключать стандартную панель инструментов, дополнительную панель инструментов или панель форматирования.



Галочка показывает, что панель видна.

Как добавлять, удалять и перемещать направляющие линии

Crystal Report Designer поддерживает направляющие линии, позволяющие точно устанавливать объекты в отчете. Направляющие линии представляют собой непечатные линии, которые, с целью выравнивания, можно устанавливать в любом месте Design и Preview Tabs. Вы можете сделать направляющие линии видимыми и невидимыми двумя способами:

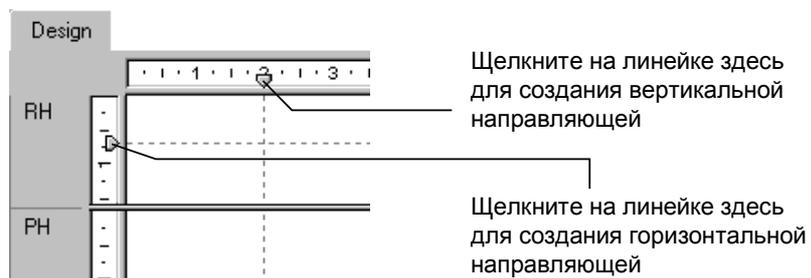
- С помощью двух команд в меню View: Guidelines in Design и Guidelines in Preview. Смотрите Guidelines in Design command и Guidelines in Preview command в оперативной Подсказке Crystal Report Designer.
- С помощью двух опций, Show Guidelines in Design и Show Guidelines in Preview в Layout Tab диалогового окна File Options. Смотрите File Options dialog box в оперативной Подсказке Crystal Report Designer.

Crystal Report Designer вставляет направляющие линии автоматически в следующих ситуациях:

- Когда Вы вставляете поле или поле формулы в свой отчет, программа автоматически создает направляющую линию у левой границы рамки поля и "привязывает" к ней поле и заголовок поля.
- Если Вы резюмируете поле, то, для обеспечения правильного выравнивания, программа привязывает полученную сводку к той же направляющей линии.
- Когда Вы щелкаете правой кнопкой мышки по затененной области слева от раздела и выбираете команду Arrange lines, программа автоматически создает в разделе одну или несколько горизонтальных направляющих линий и "привязывает" к ним поля.

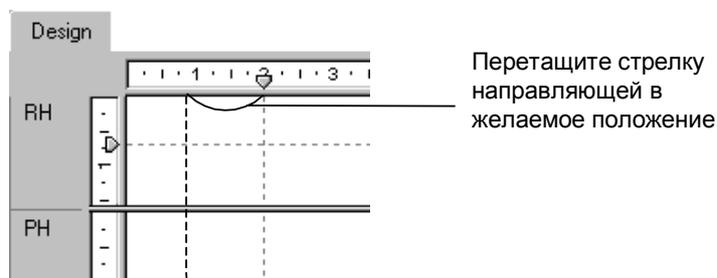
Добавление направляющих линий

Иногда Вам может потребоваться вставить направляющие линии вручную.



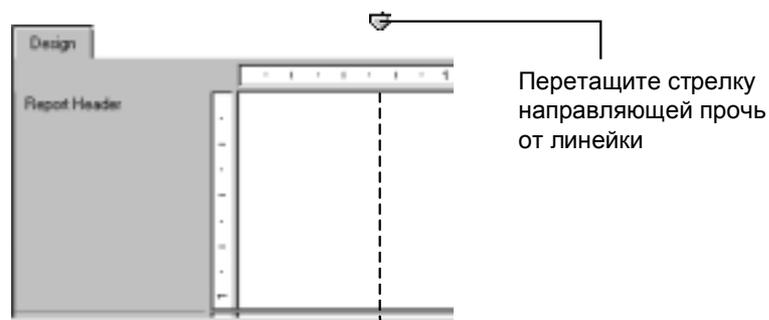
Перемещение направляющих линий

Каждая направляющая линия присоединена к стрелке на линейке, от которой она проведена.



Удаление направляющих линий

Каждая направляющая линия присоединена к стрелке на линейке, от которой она проведена.



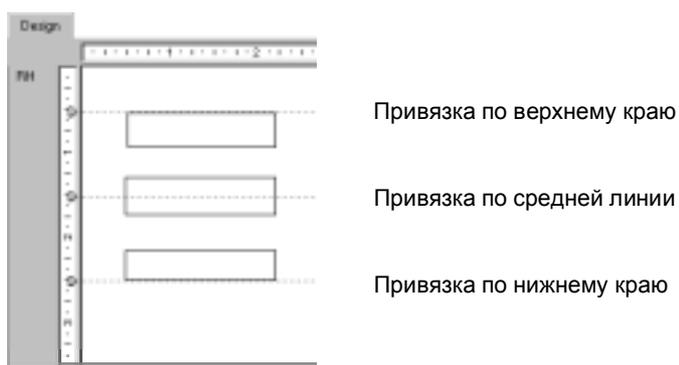
ВНИМАНИЕ: Если подключена опция *Snap to Grid*, Вы можете только вставлять и перемещать направляющие линии по сетке. Для получения всех возможностей, связанных с направляющими линиями, отключите опцию *Snap to Grid*, используя *Layout Tab* в диалоговом окне *File Options*. Смотрите *File Options dialog box* в оперативной Подсказке *Crystal Report Designer*.

Как перемещать и устанавливать объекты с использованием направляющих линий

Установка объектов с помощью направляющих линий

Направляющие линии обладают свойством привязки, которое автоматически привязывает к ним объекты. Это означает, что объекты будут присоединяться к направляющим линиям как к устройствам управления курсором.

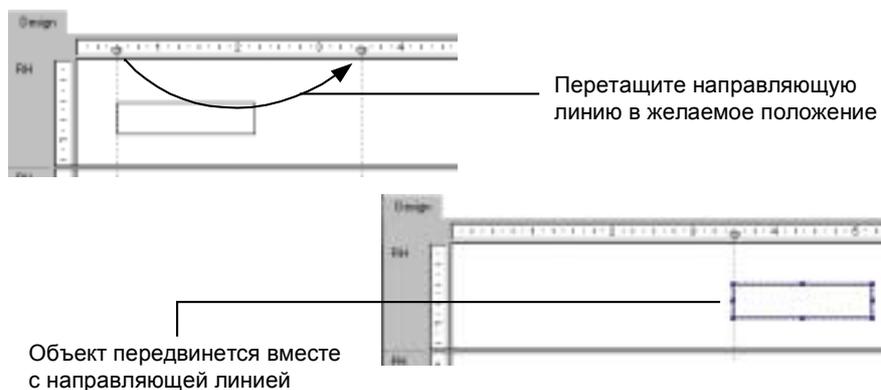
Объект может быть привязан к горизонтальной направляющей линии по его верхнему краю, нижнему краю или по срединной горизонтальной линии (линии, разделяющей объект горизонтально на две части).



Объект может быть привязан к вертикальной направляющей линии по его правому краю, левому краю или по срединной вертикальной линии (невидимой линии, разделяющей объект вертикально на две части).



Если свойство привязки к направляющей линии подключено, и Вы начинаете перемещать направляющую линию, то все привязанные к ней объекты также будут перемещаться.



ВНИМАНИЕ: Если Вы перемещаете направляющую линию, то с ней перемещается любой привязанный к ней объект. Но если Вы перемещаете объект, привязанный к направляющей линии, то программа не будет перемещать саму направляющую линию.

Изменение размеров объектов с использованием направляющих линий

Если Вы привяжете один или несколько объектов к направляющим линиям с двух сторон (сверху и снизу, или справа и слева), то Вы сможете изменить размеры этих объектов путем перемещения одной или обеих этих направляющих линий. Это может быть особенно полезно, когда Вам необходимо одновременно одинаково изменить размеры нескольких объектов.

Данный процесс состоит из двух шагов:

1. Сначала Вы должны привязать объект к двум направляющим линиям.
2. Затем, путем перемещения направляющей линии, Вы должны изменить размер объектов.

Привязка объектов к двум направляющим линиям

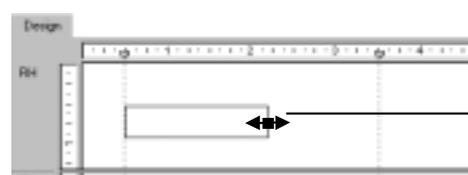
Возможно, Вам потребуется изменить размеры двух или более объектов, которые могут быть одинакового или различного размера (по высоте или ширине). Процесс изменения размера с помощью двух направляющих линий будет одинаковым в обоих случаях.

1. Начните с создания направляющей линии и привязки к ней одной стороны объекта.



2 Создайте вторую направляющую линию справа от объекта. Она не должна касаться объекта.

3. Теперь щелкните по объекту мышкой, чтобы активизировать ручки изменения размера.



4 Перетащите ручку изменения размера ко второй направляющей линии.

Произойдет привязка объекта ко второй направляющей линии.

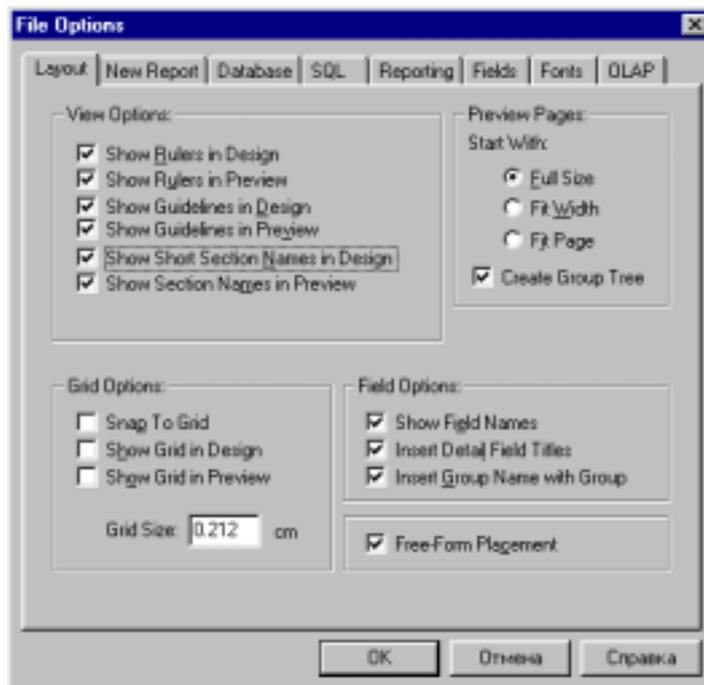


5. Повторяйте шаги 2-4 для каждого следующего объекта, который Вы хотите привязать к обеим направляющим линиям.
6. Если объекты имеют не тот размер, который Вам нужен, перемещайте одну или обе направляющие линии до тех пор, пока не достигнете требуемого размера.

Как подключать/отключать сетку

Design и Preview Tabs имеют структуру базовой сетки, которую Вы можете активизировать и изменить ее размер в диалоговом окне File Options.

1. Выберите из меню File команду **Options**. На экране отобразится диалоговое окно File Options с активной закладкой Layout.



2. Установите в данном диалоговом окне опции Grid Size, Snap to и Show.

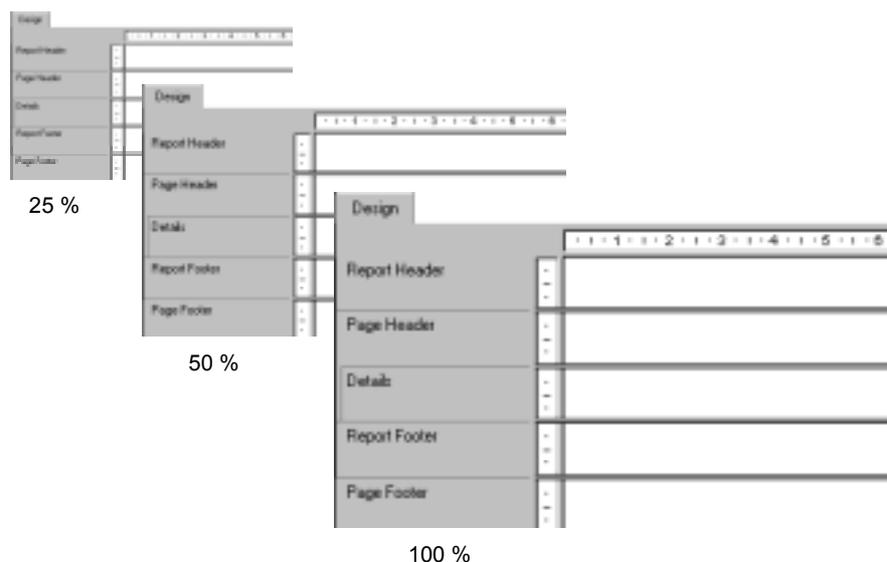
ВНИМАНИЕ: По умолчанию все опции сетки отключены.

Как увеличивать и уменьшать изображение отчета

Вы можете легко увеличить изображение своего отчета. Можно выбрать любое процентное увеличение в пределах от 25% до 400%. Данное свойство активно как в Design Tab, так и в Preview Tab.

Для уменьшения или увеличения изображения отчета установите требуемый Вам уровень уменьшения (увеличения) с помощью выпадающего меню **Magnification Factor**, расположенного на стандартной панели инструментов.

Установите желаемый коэффициент увеличения и нажмите ОК после завершения.



Для получения более точного общего представления о внешнем виде Вашего отчета полезнее просматривать его при небольшом увеличении изображения. Большое увеличение изображения полезно использовать при внимательном рассмотрении деталей отчета.

Как отменять/восстанавливать действия

Crystal Report Designer включает много уровней отмены. С их помощью Вы можете пошагово отменять последовательные изменения до тех пор, пока не достигнете требуемого состояния отчета.

Кроме того, программа может восстанавливать ранее отмененные свойства. Если, например, Вы переместили объект, и Вам не понравилось его новое положение, то Вы можете щелкнуть мышкой по **Undo**, восстановив таким образом объект в его первоначальное положение. Если после этого Вы вдруг изменили решение, то можете щелкнуть мышкой по **Redo**, и объект снова вернется в измененное положение.

- Для отмены действия, щелкните по кнопке Undo на стандартной панели инструментов. После первого нажатия на кнопку произойдет отмена самого последнего изменения, внесенного в отчет. Каждое последующее нажатие на Undo будет вызывать отмену предыдущего изменения.
- Для восстановления изменения после его отмены, щелкните по кнопке Redo на стандартной панели инструментов.

При отсутствии изменений, которые можно отменить/восстановить, или если Вы произвели изменения, которые нельзя отменить, программа блокирует кнопку **Undo** и команды **Undo/Redo**.

ВНИМАНИЕ: Если Вы отменили какое-либо действие (например, Действие А), а затем произвели некоторое новое действие, то Вы уже не сможете восстановить действие А.

Как детализировать суммированные данные

С помощью курсора "drill down" Вы можете детализировать данные (показывать данные, входящие в отдельные группы). Смотрите: *Sorting, Grouping, and Totaling*.

Курсор детализации (drill down cursor)

Crystal Report Designer позволяет Вам детализировать групповую или суммарную информацию в Preview Tab как в Стандартном виде, так и в виде Группового дерева (см. *Стандартный вид* и *Вид Группового дерева*). Если Вы установите курсор на какое-либо суммарное значение, составляющие которого Вам необходимо получить, то программа отобразит курсор детализации.

Если Вы дважды щелкните правой кнопкой мышки, программа покажет детали, входящие в данное суммарное значение. Например, если курсор детализации расположен на карте города, то, дважды щелкнув мышкой по определенному месту карты, Вы сможете получить о нем детальную информацию. Смотрите: *Курсоры*.

- Если у Вас только одно суммарное значение, то Вы можете посмотреть суммарное значение или данные из отдельных записей, которые были суммированы.
- Если у Вас несколько суммарных значений, то Вы можете посмотреть суммарные значения, входящие в данное суммарное значение (например, карты городов, входящие в карты регионов), или данные из отдельных записей, которые были суммированы.

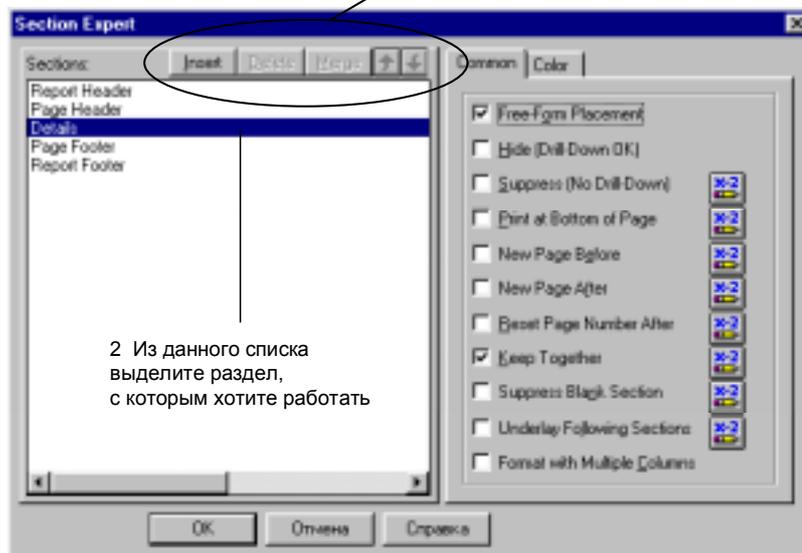
ПРАКТИКА (Разделы и области)

Как добавлять, удалять, перемещать и объединять разделы

Изначально Design Tab открывается со стандартным набором разделов отчета, каждый из которых имеет стандартный размер (см. *Области Design Tab*). Вы не можете удалять эти оригинальные разделы, но можете добавлять к ним дополнительные. Если Вы добавили раздел, то далее Вы можете удалять его, перемещать относительно других аналогичных разделов или объединять со связанными с ним разделами.

Для осуществления этих операций существует несколько способов, простейшим среди которых является использование кнопки **Section Expert**.

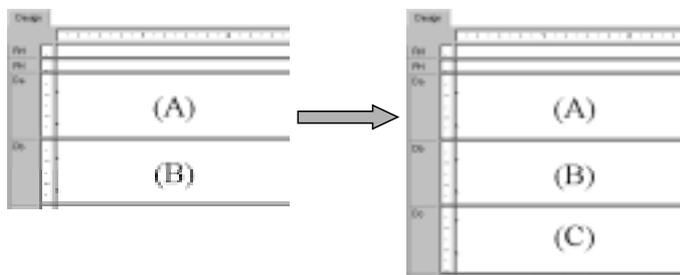
1. Щелкните по кнопке **Section Expert** на стандартной панели инструментов. На экране отобразится окно Section Expert со списком всех разделов отчета. При наличии более одного раздела какого-либо вида, подобные разделы будут обозначены А, В, С и т.д.



Вставка раздела

Щелкните мышкой по кнопке **Insert**. Программа добавит новый раздел непосредственно под высвеченным разделом.

ВНИМАНИЕ: Вы также можете вставить раздел, щелкнув правой кнопкой мышки по затененной области слева от любого раздела в *Design* или *Preview Tabs* и выбрав из появившегося меню горячих клавиш команду *Insert Section Below*.

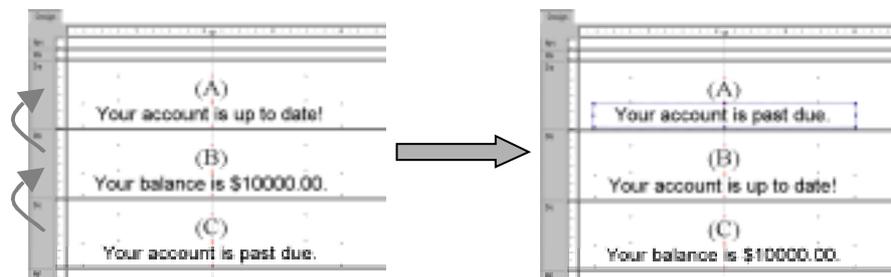


Удаление раздела

Щелкните мышкой по кнопке **Delete**. Программа удалит из Вашего отчета выделенный раздел.

Перемещение раздела

1. Выделите раздел, который хотите переместить.
2. Для перемещения раздела щелкните по стрелкам *Up* или *Down*.



3 Выделив раздел (C), дважды щелкните по стрелке *Up*.

Данные, изначально находившиеся в разделе (C), переместятся в раздел (A). Данные из всех остальных разделов переместятся вниз.

ВНИМАНИЕ: Вы можете перемещать разделы вверх и вниз только в пределах области.

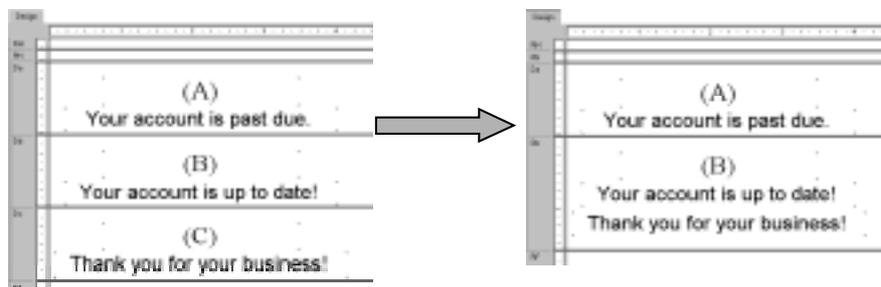
ВНИМАНИЕ: Буквы, идентифицирующие разделы, описывают их относительное (в противоположность оригинальному) положение. Следовательно, если Вы переместите раздел "C" вверх, он станет разделом "B". Теперь он уже не будет обозначен как "C".

Объединение двух соотнесенных разделов

Иногда, установив объекты в два раздела (когда они печатаются последовательно), Вам может потребоваться объединить их в один раздел (когда они печатаются одновременно). Вы можете объединить эти два раздела в один, а затем переорганизовать объекты в полученном новом разделе, если это необходимо.

1. Переместите разделы так, чтобы два раздела, которые Вы хотите объединить, шли в списке последовательно.

2. Выделите верхний раздел.
3. Щелкните мышкой по кнопке **Merge**. Программа объединит оба раздела в один.
4. Переорганизуйте объекты, если это необходимо.



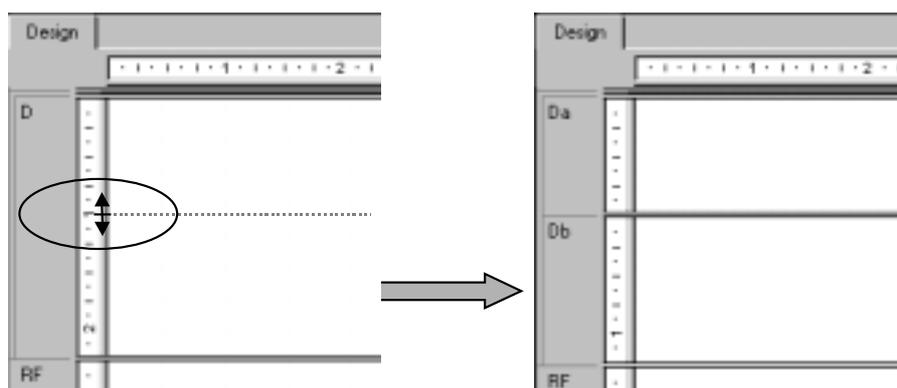
5 Выделив раздел (C), щелкните по кнопке Merge. Раздел (C) объединится с разделом (B) в один раздел.

Как разбивать разделы и изменять их размеры

В Design Tab Вы без труда можете разбивать раздел на два или более разделов или изменять размеры разделов.

Разбиение раздела

1. Установите указатель на левую границу раздела, который Вы хотите разбить.
2. Когда указатель превратится в курсор разбиения раздела, щелкните по границе. Смотрите: *Курсоры*.
3. Когда появится горизонтальная линия, переместите ее вверх или вниз до того места, в котором Вы хотите разбить раздел.

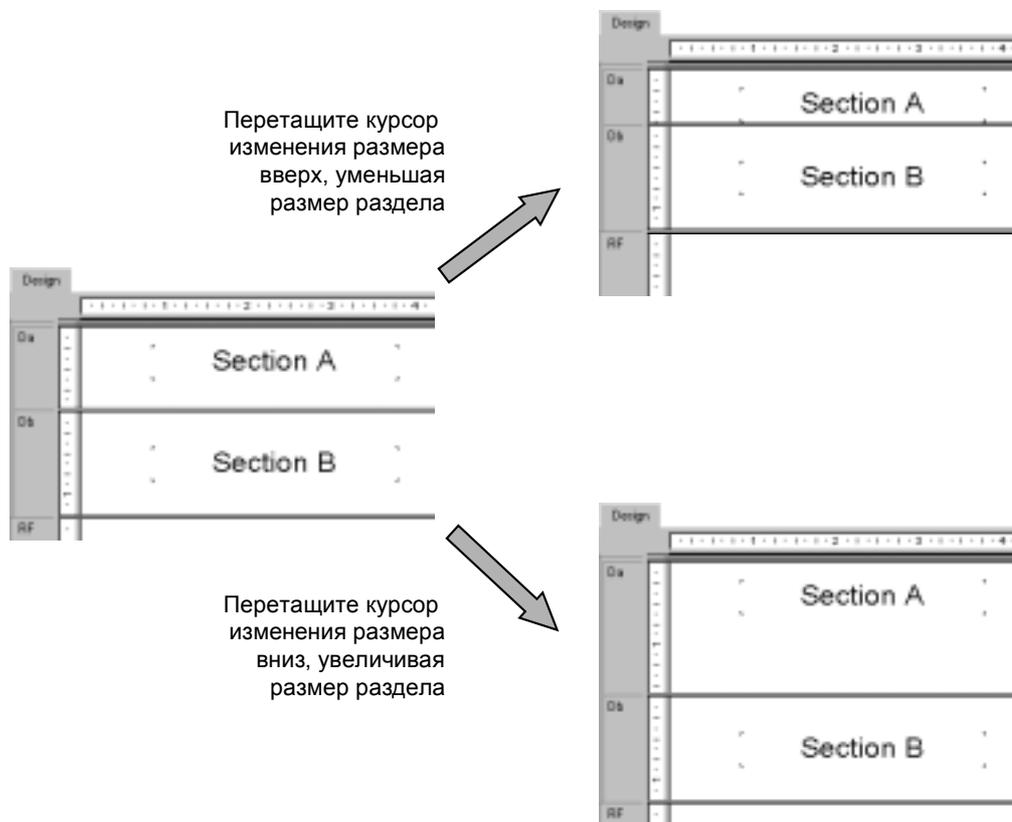


4 При появлении курсора разбиения раздела переместите горизонтальную линию в желаемое положение

Раздел разобьется по линии

Изменение размера раздела

Переместите указатель на нижнюю границу раздела, размер которого Вы собираетесь изменить, и, когда указатель превратится в курсор изменения размера, передвиньте границу, увеличивая или уменьшая таким образом размер раздела.



Если у Вас в разделе установлены один или несколько объектов, и Вы хотите изменить размер раздела, чтобы убрать ненужные пустые места, щелкните правой кнопкой мышки по затененной области слева от раздела в Design и Preview Tabs и выберите из появившегося меню горячих клавиш команду **Fit Section**. Программа автоматически изменит размер раздела, переместив его нижнюю границу к основанию нижнего объекта в разделе.

Раздел автоматически увеличивается в размере по вертикали в двух случаях:

1. когда Вы устанавливаете объект, и этот объект оказывается больше (по вертикали) раздела, в который Вы его устанавливаете.
2. когда Вы раздвигаете объект (по вертикали) так, что он становится больше раздела, в котором находится.

ВНИМАНИЕ: *Невозможно изменить размер раздела так, чтобы он стал меньше находящегося в нем объекта.*

РАБОТА С БАЗАМИ ДАННЫХ

Обзор баз данных

Хотя существуют сотни Систем управления базами данных (DBMS), Crystal Report Designer помогает исключить множество существующих между ними различий, т.к. он присоединен к фактическим файлам баз данных. Процесс работы с файлами, таблицами, полями и записями баз данных почти одинаков, независимо от фактического типа данных, доступ к которым осуществляется.

В настоящем разделе описано несколько концепций и задач, общих для работы с файлами базы данных. Использование псевдонимов, расположение перемещенных или переименованных файлов базы данных, работа с индексными таблицами и связывание таблиц – все это моменты, с которыми приходится сталкиваться каждому, кто создает отчеты в Crystal Report Designer. Раздел *Использование SQL и баз данных SQL* имеет особое значение для тех, кто осуществляет доступ к данным в базах данных SQL, а также базам данных в других форматах, доступ к которым осуществляется через ODBC.

Основы реляционной модели баз данных

Наиболее популярная архитектура файлов в базе данных основана на реляционной модели. Таким образом, приложения, позволяющие Вам создавать базы данных с помощью реляционной модели, часто обращаются к Реляционным системам управления базами данных (RDBMS).

В реляционной базе данных данные организованы в систему рядов и колонок. Ряды называются записями, а колонки полями. Каждая запись содержит набор взаимосвязанных данных, вся информация в которых относится, например, к определенному заказчику. Каждое поле обращается к общему типу данных, присутствующему во всех записях, например, имена заказчиков. Записи и поля хранятся в таблице базы данных. Приведенная ниже диаграмма иллюстрирует базовую реляционную модель базы данных:

Customer ID	Customer Name	Address 1	City	Region	Postal Code
1	Bike-O-Rama	7464 St.Georges Way	Sterling Heights	MI	48358
2	The Pedellars Inc.	410 Eighth Avenue	Dekalt	IL	60148
3	Bikes 'R Us Enterprises	7429 First Boulevard	Blacklick	OH	4300
4	Cycle Sporin Corporation	8287 St.Georges Way	Huntsville	AL	35818
5	Sporting Wheels Inc.	480 Grant Way	San Diego	CA	92160
6	The Cytclists Company	1984 Sydney Street	Austin	TX	78770
7	Ride 'Em Cowboy Corp.	8194 Pender Avenue	Eden Prairie	MN	55360
8	XYZ Enterprises	3802 Georgia Court	Des Moines	LA	5030
9	Trail Blazer's Place	6938 Second Street	Madison	WI	53795
10	The Cyclists Incorporated	4861 Second Road	Newbury Park	CA	9134

Колонка

Ряд

Часто данные, находящиеся в двух различных таблицах, могут соотноситься друг с другом по общему полю. Например, таблица Заказчики будет иметь идентификатор заказчика для каждого заказчика, и таблица Заказы будет иметь идентификатор заказчика для каждого разместившего заказ. Это отражает взаимоотношения между таблицами. Две таблицы могут быть связаны (см. *Связывание таблиц*) по общему полю. Чтобы понять, какие отношения могут устанавливаться между двумя таблицами, изучите приведенную ниже диаграмму.

Customer Table					
Customer ID	Customer Name	Address 1	City	Region	Postal Code
1	Bike-O-Rama	7464 St.Georges Way	Sterling Heights	MI	48358
2	The Pedellars Inc.	410 Eighth Avenue	Dekalt	IL	60148
3	Bikes 'R Us Enterprises	7429 First Boulevard	Blacklick	OH	4300
4	Cycle Sporin Corporation	8287 St.Georges Way	Huntsville	AL	35818
5	Sporting Wheels Inc.	480 Grant Way	San Diego	CA	92160
6	The Cytclists Company	1984 Sydney Street	Austin	TX	78770
7	Ride 'Em Cowboy Corp.	8194 Pender Avenue	Eden Prairie	MN	55360
8	XYZ Enterprises	3802 Georgia Court	Des Moines	LA	5030
9	Trail Blazer's Place	6938 Second Street	Madison	WI	53795
10	The Cyclists Incorporated	4861 Second Road	Newbury Park	CA	9134

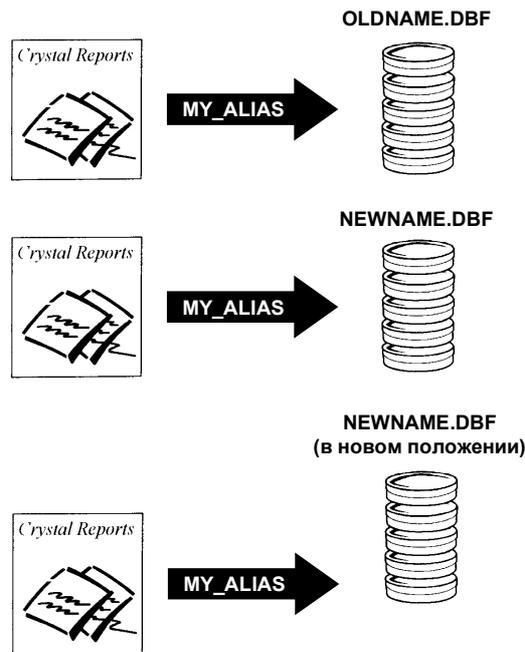
Order Table				
Customer ID	Order ID	Order Amount	Order Data	Ship Via
1	1	\$41.90	94/12/02	UPS
41	1001	\$5,060.27	94/12/02	Pickkup
77	1003	\$186.86	94/12/02	UPS
18	1004	\$823.05	94/12/02	Pickup
64	1005	\$0.00	94/12/03	Loomis
7	1005	\$64.90	94/12/03	Purolator
32	1007	\$49.50	94/12/03	Parcel Post
11	1008	\$2,214.93	94/12/03	Purolator
25	1009	\$29.00	94/12/03	Loomis
34	1010	\$14,872.28	94/12/03	UPS
71	1011	\$0.00	94/12/03	Purolator
66	1012	\$10,259.10	94/12/03	Loomis
28	1013	\$1,142.12	94/12/03	Parcel Post
8	1014	\$0.00	94/12/04	Purolator
72	1015	\$0.00	94/12/04	UPS
64	1016	\$563.70	94/12/04	FedEx
38	1017	\$72.00	94/12/05	Purolator
37	1018	\$115.50	94/12/05	Loomis
30	1019	\$0.00	94/12/05	Parcel Post
25	1020	\$67.80	94/12/05	FedEx

Отношения между таблицами, основанные на общем поле

Псевдонимы

По различным причинам имена и местоположение в базе данных могут изменяться. Если Вы создаете отчет, затем изменяете имя или местоположение таблицы или файла, то Crystal Report Designer должна иметь возможность нахождения нового имени или местоположения. Это имеет особое значение, если Вы создаете в отчете формулы, которые должны иметь доступ к переименованной или перемещенной таблице. Установить ссылку для одного поля было бы не сложно, но значительно сложнее будет выявить каждую формулу, использующую данное поле.

Для решения данной проблемы Crystal Report Designer использует псевдонимы для обращения к таблицам и файлам базы данных. Псевдонимы представляют собой указатели, внутренние устройства, сообщающие программе, где искать поле базы данных. Теперь, если Вы измените имя или местоположение базы данных, Вам просто нужно будет восстановить указатель. Смотрите: *Размещение файлов*. Имя псевдонима не изменяется, поэтому Ваши формулы не будут затронуты. Crystal Report Designer, для получения имени и местоположения, ищет псевдоним, переходит в новое местоположение поля базы данных, после чего выполняет формулу безо всяких проблем.

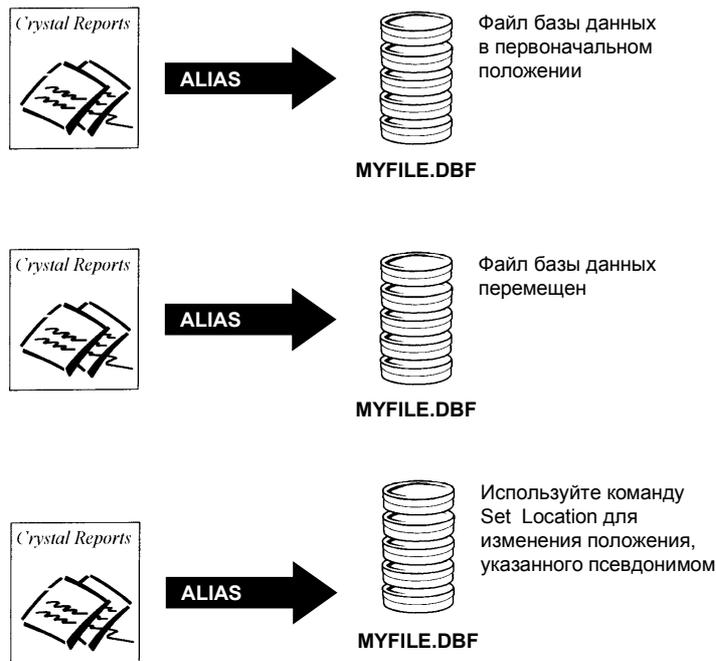


Crystal Report Designer автоматически присваивает имена псевдонимов по умолчанию таблицам базы данных, когда Вы в первый раз выбираете таблицу или файл. По умолчанию, псевдоним соответствует оригинальному имени таблицы. В базах данных, где таблица базы данных представляет отдельный файл (например, dBASE), имя файла базы данных используется без расширения имени файла. Например, если Вы используете файл базы данных dBASE, COMPANY.DBF, то программа присвоит по умолчанию файлу имя псевдонима COMPANY. Вы можете принять псевдоним по умолчанию или присвоить таблице базы данных новый псевдоним.

В любое время Вы можете изменить псевдоним с помощью команды **Set Alias** из меню Database. Однако, если Вы уже создали в своем отчете формулы, использующие оригинальное имя псевдонима, то Вам придется отредактировать эти формулы, чтобы они использовали новый псевдоним.

Размещение файлов

Если файл базы данных был перемещен или переименован, то в следующий раз при печати отчета Crystal Report Designer не сможет найти данные. В других случаях, отчет может создаваться на одной машине, где в определенной директории хранятся все данные базы данных; затем отчет копируется или переносится на другую машину, которая хранит эти же данные, но в другой директории. В любом из этих случаев Вам необходимо проверить местоположение файлов базы данных, доступ к которым осуществляет отчет, и восстановить указатели псевдонимов на новое местоположение или имя базы данных.



Команда **Verify Database** из меню Database проверяет указатели псевдонимов, сохраненные в файле отчета, чтобы удостовериться в том, что эти файлы хранятся в указанных директориях. Если база данных не обнаружена в заданном месте, программа уведомит Вас об этом несоответствии. Смотрите *Verify Database command* в оперативной Подсказке Crystal Report Designer.

Используйте команду **Set Location** из меню Database для изменения указателя псевдонима, сохраненного Crystal Report Designer. Команда **Set Location** обеспечивает простой способ указания нового имени или местоположения файлов базы данных. Кроме того, данная команда может использоваться для изменения источника данных ODBC, используемого отчетом. Смотрите: *Как изменять источник данных ODBC, к которому отчет осуществляет доступ*, а также *Set Location command* в оперативной Подсказке Crystal Report Designer.

Индексные таблицы

Создание индексов для таблиц базы данных может увеличить скорость доступа к данным и уменьшить время, необходимое программе для оценки данных. Некоторые приложения DBMS автоматически индексируют таблицы базы данных, а другие требуют, чтобы Вы создавали индекс сами. Для наиболее эффективной генерации отчета удостоверьтесь, чтобы каждая таблица Вашей базы данных имела соответствующий индекс.

ВНИМАНИЕ: *Некоторые приложения DBMS не поддерживают индексные таблицы. Обратитесь к документации для Вашей DBMS, чтобы выяснить, поддерживает ли она индексы, и как их создавать. Если в документации индексные таблицы не упоминаются, то, возможно, данная DBMS их не поддерживает; в этом случае Вы должны будете связывать таблицы на основе общих полей. Visual Linking Expert также может помочь Вам определить, включают ли Ваши таблицы индексы. Смотрите Visual Linking Expert в оперативной Подсказке Crystal Report Designer.*

Индексы организуют записи в таблице реляционной базы данных так, чтобы максимально упростить размещение данных. Например, предположим, что Ваша таблица содержит следующие данные:

Order#	Customer	Amount
10444	Allez Distribution	25141.50
10470	BG Mountain Inc.	19164.30
10485	Sierra Mountain	8233.50
10488	Mountain Toad	24580.50
10495	SFB Inc.	7911.80
10501	La Bomba de Bicicleta	1956.20
10511	BG Mountain Inc.	1683.60
10544	Sierra Bicycle Group	19766.20
10568	Mountain Tops Inc.	29759.55
10579	Sierra Bicycle Group	12763.95

Информация в данной таблице организована в соответствии с полем Order#. Это удобно, если Вы хотите найти в данной таблице информацию по номеру заказа. А если Вам потребуется найти всю информацию, относящуюся к определенному заказчику?

Скажем, Вы хотите найти все заказы, сделанные фирмой Sierra Bicycle Group. Процесс поиска в базе данных должен начаться с нахождения первого номера заказа в списке и проверки имени заказчик на предмет соответствия его запрошенному. Если найденное имя не соответствует запрошенному, то происходит переход ко второму номеру заказа, где повторяется та же процедура. Когда обнаруживается номер заказа, имя заказчика для которого соответствует запрошенному, происходит поиск необходимой информации, а затем – переход к следующему номеру заказа. При использовании такой технологии для каждой отдельной записи в таблице должны считываться оба поля: Order# и Customer. И, следовательно, для подобного просмотра огромных таблиц баз данных с тысячами и даже миллионами записей потребуется очень много времени и компьютерной обработки.

Вместо этого, Вы можете создать индекс для таблицы на основе поля Customer. Такой индекс мог бы выглядеть следующим образом:

Customer	Pointer to Order#
Allez Distribution	10444
BG Mountain Inc.	10470
BG Mountain Inc.	10511
La Bomba de Bicicleta	10501
Mountain Toad	10488
Mountain Tops Inc.	10568
SFB Inc.	10495
Sierra Bicycle Group	10544
Sierra Bicycle Group	10579
Sierra Mountain	10485

В данном индексе информация организована по имени заказчика, а не по номеру заказа. Кроме того, следует отметить, что вторая колонка, фактически, содержит указатели на отдельные номера заказов из оригинальной таблицы. Используя данный индекс, механизм базы данных может искать информацию в колонке Customer до тех пор, пока не обнаружит интересующего нас заказчика, в данном случае Sierra Bicycle Group.

Для каждого требуемого элемента заказчика, обнаруженного в индексе, механизм базы данных ищет в таблице соответствующий заказ по указателю во второй колонке индекса. Считываются только заказы, сделанные интересующими заказчиками. И, наконец, т.к. информация в индексе организована по именам заказчиков, механизму базы данных не придется продолжать поиск в индексе или таблице после того, как будет обнаружен элемент индекса, не соответствующий требуемому заказчику.

Результатом такого высокоорганизованного поиска по индексу в таблице базы данных является значительное повышение скорости. Использование индексов ускоряет поиск данных и генерацию отчета, что имеет особое значение при создании отчетов по файлам большой базы данных.

Связывание таблиц

Связь между таблицами устанавливается таким образом, чтобы записи из одной таблицы соотносились с соответствующими записями из другой таблицы. Например, если Вы работаете с таблицей Заказы и таблицей Заказчики, то Вы связываете эти таблицы так, чтобы каждый заказ (из таблицы Заказы) соотносился с разместившим его заказчиком (из таблицы Заказчик).

При установке связи используется поле, общее для двух таблиц. Crystal Report Designer использует связь для соотнесения записей в одной таблице с записями в другой таблице. В рассматриваемом примере связь обеспечивает принадлежность данных в каждом ряду отчета к одному заказу.

Исходящая и входящая связь

Когда Вы связываете две таблицы, то связь устанавливается *от* одной таблицы *к* другой. Вы должны это четко представлять. Исходящая таблица используется как первичная, а входящая таблица - как поисковая, в которой исходящая таблица ищет записи. При простой связи Crystal Report Designer смотрит первую запись в исходящей таблице и находит все соответствующие записи в поисковой таблице. После того, как в поисковой таблице выявлены все соответствия для первой записи из исходящей таблицы, осуществляется аналогичный поиск соответствий для второй записи в исходящей таблице.

ВНИМАНИЕ: *Crystal Report Designer может связать две записи на основе частичного соответствия данных строки. Это называется частичной связью. Для разрешения частичной связи в Crystal Report Designer подключите флажок Allow partial text matches в диалоговом окне Link Options. В качестве примера частичной связи можно привести следующий: запись со значением поля "Chris" можно связать с записью со значением поля "Christopher". Однако, частичная связь работает только в тех случаях, когда значение в поисковой таблице длиннее значения в исходящей таблице. Другими словами, значение "Chris" может быть связано со значением "Christopher", но не наоборот. Смотрите Link Options dialog box в оперативной Подсказке Crystal Report Designer.*

Отношения записей при связи

Если Вы связываете записи из одной таблицы с другой таблицей, то между этими записями устанавливаются отношения одного из двух типов:

1. один-к-одному
2. один-ко-многим.

Отношения один-к-одному

При таких отношениях между записями в двух связанных таблицах для каждой записи в первичной таблице имеется только одна соответствующая запись в поисковой таблице (на основе связанных полей). Например, в базе данных CRAZE.MDB, таблица Employee (Служащие) может быть связана с таблицей Employee Addresses (Адреса служащих) по полю Employee ID, имеющемуся в обеих таблицах. Таблица Employee содержит информацию о служащих компании, их должностях, окладах и условиях найма на работу. Таблица Employee Addresses содержит домашние адреса всех служащих. В каждой из этих таблиц для каждого служащего имеется только одна запись. Следовательно, если таблица Employee связана с таблицей Employee Addresses, то для каждой записи в первой таблице будет найдена только одна запись во второй таблице. Такие отношения называются один-к-одному.

Отношения один-ко-многим

При таких отношениях между записями в двух связанных таблицах для каждой записи в первичной таблице имеется более одной соответствующей записи в поисковой таблице (на основе связанных полей). В базе данных CRAZE.MDB, таблица Customer (Заказчик) может быть связана с таблицей Orders (Заказы) на основе поля Customer ID, присутствующего в каждой таблице. Таблица Customer содержит информацию о каждом заказчике, который разместил заказ в данной компании. Таблица Orders содержит информацию о заказах, размещенных заказчиками. Так как заказчики могут сделать более одного заказа, в таблице Orders может присутствовать более одной записи для каждой записи в таблице Customer. Такие отношения называются один-ко-многим.

Вопросы быстроедействия при связях один-ко-многим

Информация, описанная в данном разделе, предназначена, чтобы помочь Вам максимально ускорить обработку и снизить сетевой трафик при создании отчета. Вы узнаете наилучшие способы использования формул выбора и индексов при связях один-ко-многим, которые повысят эффективность работы по созданию отчетов. Если Вы не воспользуетесь информацией, представленной в этом разделе, то, возможно, при создании отчета Вам придется обрабатывать десятки или даже сотни ненужных записей.

Если между двумя таблицами базы данных установлена связь один-ко-многим, и программа сравнивает записи из этих таблиц, то существует ряд факторов, определяющих, сколько записей считывает и оценивает программа. Понимание данной информации поможет Вам избежать таких ситуаций, когда требуется очень много времени на обработку или очень велик сетевой трафик.

Приведенные ниже таблицы показывают влияние различных факторов на количество записей, которое приходится считывать программе. Таблицы основаны на следующих предположениях:

- Таблица А содержит 26 записей (по одной для каждой буквы алфавита).
- Таблица В содержит 2600 записей (по 100 соответствующих записей для каждой записи в Таблице А).
- По сценарию необходимо создать отчет, который обнаружит две записи в Таблице А и 200 (100+100) соответствующих им записей в Таблице В.
 - При наилучшем сценарии, для решения задачи программе придется прочесть только 200 записей.
 - При наихудшем сценарии, для решения задачи программе придется прочесть 67,600 записей.

ВНИМАНИЕ: Вопросы производительности для файлов данных отличаются от аналогичных вопросов для баз данных SQL. Файлом данных является любая база данных не SQL, доступ к которой осуществляется непосредственно из Crystal Report Designer. В данном описании базой данных SQL является любая база данных SQL, доступ к которой осуществляется непосредственно из Crystal Report Designer или через ODBC, а также все остальные типы баз данных, доступ к которым осуществляется через ODBC. Для дополнительной информации о различиях между базами данных с прямым доступом и источниками данных ODBC смотрите: *Data Sources*.

Расширенные описания колонок таблиц

В таблицах производительности использованы следующие колонки:

- Linking or Subreport (Связь или Подотчет)
Создаете ли Вы отчет из связанных баз данных или же вставляете подотчет и связываете его с данными в своем первичном отчете?
- Selection Formula (Формула выбора)
Включает ли Ваш первичный отчет формулу выбора записей, которая устанавливает границы диапазона на ключевом (индексном) поле в Таблице А?
- Index A (Индекс А)
Проиндексирована ли Таблица А по полю, которое Вы собираетесь использовать для соотнесения записей?
- Index B (Индекс В)
Проиндексирована ли Таблица В по полю, которое Вы собираетесь использовать для соотнесения записей?
- Reads A (Считывает А)
Какое количество записей придется считывать программе из Таблицы А для нахождения двух искомым записей?
- For each A reads in B (Для каждой из А считывает в В)
Какое количество записей придется считывать программе из Таблицы В для нахождения 200 искомым записей?
- Total Records Read (Всего считано записей)
Какое общее количество записей придется обработать программе для решения задачи?

PC Data						
Linking/ Subreport	Selection Formula	Index A	Index B	Reads A	For each A reads in B	Total Records Read
Linking	No	Yes or No	Yes	26	100 (26*100)	2600
Linking	Yes	No	Yes	26	100 (26*100)	2600
Linking	Yes	Yes	Yes	2	100 (2*100)	200
Subreport	No	No	No	26	2600 (26*2600)	67,600
Subreport	No	Yes	No	2	2600 (26*2600)	67,600
Subreport	NO	Yes	Yes	26	100 (26*100)	2600
Subreport	Yes	No	No	2	2600 (2*2600)	5200
Subreport	Yes	No	Yes	26	100 (26*100)	2600
Subreport	Yes	Yes	Yes	2	100 (2*100)	200

SQL Data				
Linking/Subreport	Selection Formula	Reads A	For each A reads in B	Total Records Read
Linking	No	26	100 (26*100)	2600
Linking	Yes	2	100 (2*100)	200
Subreport	NO	26	100 (26*100)	2600
Subreport	Yes	2	100 (2*100)	200

Работа с файлами данных

Если Вы работаете с файлами данных, то, при связывании таблиц в один отчет или при добавлении подотчетов в отчет, могут возникнуть связи один-ко-многим.

Связывание файлов данных

При связи один-ко-многим программа ищет данные из связанных файлов данных следующим образом:

- При наличии формулы выбора, программа анализирует формулу и передает, что возможно, в DLL базы данных. Обычно это информация о границах диапазона. Например, рассмотрим следующую формулу выбора записи:

```
{customer.REGION} in "CA" to "IL" AND  
Remainder ({customer.CUSTOMER ID},2) = 0
```

В данной формуле, часть перед оператором "and" содержит критерии выбора диапазона в поле Region. Область поиска должна находиться между "CA" и "IL", в алфавитном порядке. Программа передает это условие в DLL базы данных (для данных PC) или на сервер (для данных SQL). Смотрите: Record and Group Selection.

Однако вторая половина формулы требует проведения обработки в Report Engine. Она использует встроенную функцию для работы и оценки значения поля и не может проводиться в DLL базы данных или на сервере. Программа не передает это условие в DLL базы данных.

- При наличии индекса для Таблицы А и при условии, что условие выбора границы диапазона основывается на индексном поле {customer.REGION} в данном примере, программа переходит непосредственно к записи, которую она ищет в Таблице А (первая запись CA), и считывает ее.
 - Для данной записи, используя индекс Таблицы В, программа размещает соответствующую запись в Таблице В.
 - Она передает эту объединенную запись (A+B) в Crystal Report Designer, которая тестирует запись на предмет соответствия всей формуле выбора.
 - Затем она считывает вторую соответствующую запись и передает объединенную запись так же, как и первую. Затем процедура повторяется для третьей записи и т.д. до тех пор, пока не будут считаны все соответствующие записи.
 - После этого программа возвращается в Таблицу А и считывает в ней следующую запись. Нет необходимости тестировать запись на предмет соответствия ее условию CA; поле проиндексировано, и все записи расположены в алфавитном порядке. Но, при этом, программа тестирует данную запись для выяснения, не выходит ли она за пределы условия "IL" (например, не может ли следующая запись относиться к Mississippi или Tennessee?). Если запись находится в пределах заданного диапазона, программа начинает тот же процесс поиска соответствующих записей в Таблице В для этой записи из Таблицы А.
 - Данный процесс продолжается до тех пор, пока программа не разместит все входящие в заданный диапазон записи из Таблицы А и соответствующие им записи из Таблицы В.

Для обнаружения двух записей в Таблице А и 100 соответствующих им записей в Таблице В программа должна будет считать 200 записей.

ВНИМАНИЕ: Поисковая таблица в связи (Таблица В) всегда должна быть проиндексирована; в противном случае Вы не сможете связать таблицы.

- Если индекс для Таблицы А отсутствует, или же индекс присутствует, но условие выбора границ диапазона не основано на индексном поле, программа будет считывать первую же запись, которую она обнаружит.
 - Для данной записи, используя индекс для Таблицы В, программа находит первую соответствующую запись в Таблице В.
 - Она передает объединенную запись (А+В) обратно в Report Engine, который тестирует эту запись относительно всей формулы выбора.
 - Затем программа находит вторую соответствующую запись в Таблице В и передает объединенную запись обратно, затем процесс повторяется для третьей записи и т.д. до тех пор, пока не будут найдены, объединены и переданы обратно все записи из Таблицы В, соответствующие первой записи в Таблице А.
 - Затем программа переходит ко второй записи в Таблице А и начинает процесс поиска и объединения снова.

Для обнаружения двух записей в Таблице А и 100 соответствующих им записей в Таблице В программа должна будет считать 2600 записей.

Подотчеты и файлы данных

Если Ваш первичный отчет основан на Таблице А, а подотчет – на Таблице В, и записи связаны, то, в первую очередь, Вы должны учитывать следующие моменты:

- Количество имеющихся подотчетов определяется индексом и формулой выбора в первичном отчете.
 - Если Таблица А проиндексирована, а первичный отчет имеет формулу выбора, которая передает информацию о границах диапазона для индексного поля, то программа будет обрабатывать 2 подотчета.
 - Если Таблица А не проиндексирована, или если Таблица А проиндексирована, но формула выбора не передает информацию о границах диапазона для индексного поля, то программа будет обрабатывать 26 подотчетов.
- Количество записей, прочитываемое для каждого подотчета, определяется индексом для Таблицы В.
 - При наличии индекса для Таблицы В, каждый раз при обработке подотчета программа будет считывать только соответствующие (100) записи.
 - При отсутствии индекса для Таблицы В, каждый раз при обработке подотчета программа будет обязательно считывать все (2600) записи из Таблицы В.

Особенности работы с базой данных SQL

Так как при использовании данных SQL индексы не критичны, основным моментом в работе как со связанными таблицами, так и с подотчетами будет вопрос наличия в первичном отчете формулы выбора, устанавливающей границы диапазона в Таблице А. Смотрите *Связывание файлов данных*.

Связанные таблицы SQL

Если в формуле выбора заданы границы диапазона, то программа передаст эти условия на сервер.

- Если имеется формула выбора, устанавливающая границы диапазона для Таблицы А, то сервер найдет записи в этой таблице, удовлетворяющие критерию выбора (2), найдет для них соответствующие записи в таблице В (по 100 записей) и вернет 200 объединенных записей в Report Engine.
- Если формула выбора отсутствует, или если имеющаяся формула выбора не устанавливает границы диапазона для Таблицы А, то сервер будет сравнивать каждую запись в Таблице А (26) с соответствующими записями в Таблице В (по 100 записей) и вернет в Report Engine 2600 объединенных записей.

В обоих случаях, Report Engine применит затем всю формулу выбора к объединенным записям.

Подотчеты и базы данных SQL

Если Вы создаете первичный отчет из Таблицы А, а подотчет – из Таблицы В, то:

- Количество подотчетов будет определяться формулой выбора в первичном отчете.
 - При наличии формулы выбора и передачи ею пределов диапазона для Таблицы А, программа будет запускать подотчет только для тех записей, которые удовлетворяют условиям границ диапазона (2).
 - Если формула выбора отсутствует, или если она не передает границы диапазона для Таблицы А, программа будет работать с подотчетом для каждой записи в Таблице А (26).
- Каждый подотчет будет считывать одинаковое количество записей, независимо от наличия или отсутствия выбора границ диапазона для Таблицы А. В Таблице В каждый подотчет будет считывать только те записи, которые соответствуют каждой считанной записи из Таблицы А (100).

Вопросы производительности для всех отчетов

Вопрос № 1

При наличии файлов данных и баз данных SQL программа проводит анализ всей формулы выбора и передает все критерии, которые может транслировать (передать), везде, где они могут физически присутствовать в формуле. Таким образом, если формула обнаруживает критерий, который может передать, затем критерий, который она передать не может, а затем снова критерий, который она может

передать, то она будет передавать сначала первую часть, пропустит вторую и затем передаст третью.

- В случае файлов данных, программа передает все возможные критерии в трансляционный слой базы данных.
- В случае баз данных SQL, она передает все возможные критерии на сервер в форме WHERE-предложения.

При наличии исключительных ситуаций программа, как правило, может передать любую часть формулы выбора записи, которая сравнивает поле с константой. Обычно это означает, что программа может передать любой вид критерия формулы выбора, который может быть установлен в Select Expert (field equal to, one of, less than, greater than, less than or equal, greater than or equal, between, starting with или like constant).

Существуют две особые ситуации с формулами выбора, которые необходимо учитывать. В этих ситуациях в формуле выбора записей заданы несколько условий, одни из которых могут быть переданы, а другие – нет.

- **Ситуации AND**

```
( {customer.REGION} = "CA" and  
{customer.CUSTOMER ID} [3 to 5] = "777" )
```

В данной ситуации программа смотрит, может ли она передать условие перед оператором "And", а не условие после него. Так как первому условию будут соответствовать только те записи, которые отвечают второму условию, она передает первое условие, ищет набор данных, ему удовлетворяющий, а затем применяет второе условие только к найденным данным. Правило для ситуаций AND заключается в том, что программа передает все условия, которые может.

ВНИМАНИЕ: Если все условия в ситуации AND могут быть удовлетворены на сервере или в DLL базы данных, то программа передаст их все.

- **Ситуации OR**

```
( {customer.REGION} = "CA" or  
{customer.CUSTOMER ID} [3 to 5] = "777" )
```

В данной ситуации программа также смотрит, может ли она передать условие перед оператором "Or", а не условие после него. В связи с тем, что первому условию могут соответствовать записи, не отвечающие второму условию, передача первого условия не имеет смысла, т.к., исходя из нее, будет найден не полный набор данных. Другими словами, даже если программа найдет все записи, удовлетворяющие первому условию, ей все равно придется просматривать все данные в таблице (таблицах) на предмет соответствия их второму условию в Crystal Report Designer. Следовательно, вместо частичного дублирования поиска данных, программа не передает никакие условия. Она осуществляет поиск из всех данных, а затем выполняет оба теста в Crystal Report Designer. Правило для ситуаций OR заключается в том, что программа передает все тесты или не передает ни одного из них.

ВНИМАНИЕ: Если все тесты в ситуации OR могут быть проведены на сервере или в DLL базы данных, то программа передаст их все.

Вопрос № 2

Чтобы удостовериться, что программа может использовать индекс на Таблицу А для улучшения производительности, проверьте:

- наличие формулы выбора;
- наличие границ диапазона в формуле выбора в ключевом (индексном) поле в Таблице А;
- что опция *Use Indexes* подключена в диалоговом окне File Options. Смотрите *File Options dialog box* в оперативной Подсказке Crystal Report Designer.

Вопрос № 3

Если используемые Вами поля из Таблицы А не проиндексированы, но существует индексное поле, которое Вы можете использовать в своем запросе выбора записей, используйте его. Например, предположим, что Вы имеете три продукта (Продукт 1, Продукт 2, Продукт 3) и хотите определить все продажи Продукта 2 в США. Для поля Product индекс отсутствует, но имеется индекс для поля Order Date. Так как Вы знаете, что поставки Продукта 2 начались только с июля 1995, Вы можете ускорить обработку с помощью формулы выбора, ограничив свой отчет только заказами, размещенными с июля 1995 г. и позднее. В этом случае программа использует индекс Order Date для поиска только тех заказов, которые поступили с июля 1995 г. включительно (небольшая часть базы данных), и после этого ей придется уже просматривать не всю базу данных, а только небольшую ее часть на предмет наличия заказов на Продукт 2.

Visual Linking Expert

Visual Linking Expert позволяет Вам без труда связывать две или более таблицы. Если Вы выберете команду Add Database To Report из меню Database и выберете дополнительную таблицу базы данных, то на экране появится Visual Linking Expert и отобразит дополнительную таблицу базы данных. Смотрите *Visual Linking Expert* и *Add Database To Report command* в оперативной Подсказке Crystal Report Designer.

Простейшим способом связывания двух таблиц базы данных является нажатие на кнопку *Smart Linking* в Visual Linking Expert. Smart Linking автоматически выбирает связи для Ваших таблиц на основе общих полей в таблицах или индексных полей (если Ваша база данных поддерживает индексные поля).

Связывание индексных таблиц

Если Вы связываете таблицы базы данных с прямым доступом, то связь следует устанавливать из первичной таблицы к индексному полю в поисковой таблице. Связываемое поле из первичной таблицы может быть проиндексировано, но это необязательно. Однако связываемое поле из поисковой таблицы обязательно должно быть проиндексировано.

Кроме того, поля, используемые для связывания таблиц, должны содержать данные одинакового типа. Например, Вы можете связать строчное поле в одной таблице со строчным полем другой таблицы, или же связать цифровые поля в обеих таблицах; однако нельзя связывать строчное поле одной таблицы с цифровым полем в другой таблице.

ВНИМАНИЕ: Некоторые приложения DBMS позволяют конвертировать значение из поля в другой тип данных в индексе. Например, поле в таблице может быть цифровым, а индекс конвертирует значение поля в строчное. Однако, если Вы решите использовать данное поле для связи с другой таблицей, то Вы должны устанавливать связь с полем с оригинальным типом данных. Вы не можете связывать строчное значение с цифровым полем, которое было конвертировано в строчное в индексе.

ВНИМАНИЕ: Если Вы связываете таблицы из двух различных источников данных ODBC, например, с Сервера MS SQL и из Oracle, то для установления связи между таблицами Вы можете использовать только строчные поля. Строчные поля сохраняются в базах данных одинаковым образом, независимо от источника данных. Однако другие типы значений могут сохраняться в различных источниках данных по-разному, так что Вы не сможете связать разные источники данных в Crystal Report Designer, используя какие-либо другие значения, кроме строчных.

Изменение индекса, используемого при установлении связи

При применении свойства Smart Linking для связывания таблиц с использованием поля, являющегося составляющей нескольких индексов (двух или более), Crystal Report Designer выбирает для связи один индекс. Этот индекс может совпасть с тем, который Вы хотите использовать, а может и не совпасть. Для определения используемого индекса и, в случае необходимости, его изменения, используйте раздел *Index* диалогового окна Link Options.

Для вызова диалогового окна сделайте следующее:

- дважды щелкните кнопкой мышки по интересующей Вас линии связи,
- выберите интересующую Вас линию связи и щелкните мышкой по кнопке *Options* в нижней части Visual Linking Expert, или
- выберите интересующую Вас линию связи, щелкните правой кнопкой мышки и выберите из появившегося меню горячих клавиш команду **Options**.

Раздел *Index* диалогового окна Link Options состоит из двух частей:

1. Выпадающее окно *Index In Use* отображает индекс, используемый в настоящий момент. Если Вы щелкните мышкой по стрелке, то отобразится список остальных индексов, которые можно использовать для связи, а также опция *no specific index*. Если Вы используете индексированную таблицу базы данных и не находите в списке какой-либо определенный индекс, который Вы ожидали там увидеть, воспользуйтесь кнопкой *Add Index*.
2. Окно списка *Fields in Index* отображает поля, включенные в индекс, выбранный в настоящий момент в окне *Index In Use*.

Если Вы выберете опцию *no specific index*, то в следующий раз, когда Вы будете печатать отчет в Preview Tab, программа выберет для Вас индекс.

ВНИМАНИЕ: Не все приложения DBMS поддерживают индексированные таблицы. Перед выбором индексов для установления связи проверьте, использует ли Ваша база данных индексы. Для получения информации о возможности использования Вашей DBMS индексов и о том, как их создавать, обратитесь к документации для Вашей DBMS.

Методы поиска таблиц (базы данных с прямым доступом)

Если одна таблица связана с двумя или более таблицами, Crystal Report Designer должен знать, в каком порядке он должен осуществлять поиск и связывать данные из первичной таблицы с данными во второй, третьей, четвертой и т.д. поисковой таблице.

Crystal Report Designer предлагает три различных варианта поиска данных в двух или более поисковых таблицах из одной первичной таблицы:

1. Одновременный поиск в обеих таблицах.
2. Поиск всех соответствий в одной, а затем в другой таблице (A - B, A - C).
3. Поиск всех комбинаций двух файлов.

ВНИМАНИЕ: Эти опции недоступны при использовании данных из источника данных ODBC.

Перечисленные опции доступны, если только у Вас есть одна первичная таблица, которая связана с двумя или более поисковыми таблицами. В каждой из связей, связь должна устанавливаться из первичной таблицы. Например, перечисленные поисковые опции будут доступны, если Вы устанавливаете связи из таблицы Customer с таблицей Orders и из таблицы Customer с таблицей Credit. Однако, они будут недоступны при установлении связей из таблицы Customer с таблицей Orders и из таблицы Credit с таблицей Customer.

В рассматриваемом примере, для каждого демонстрируемого метода будут показаны три поля из трех различных связанных таблиц. В каждом случае таблица Customer связана с таблицей Orders и таблицей Credit. Будут показаны следующие поля: поле Customer Name из таблицы Customer, поле Amount из таблицы Credit и поле Order Amount из таблицы Orders. Эти поля не обязательно должны быть полями связи таблиц, но данные в этих полях иллюстрируют, как осуществляется поиск данных при использовании каждого из трех методов поиска.

Одновременный поиск в обеих таблицах

Для каждой записи в таблице Customer данная опция ищет соответствующую запись в таблице Credit и в таблице Orders. Затем она ищет следующую соответствующую запись в таблице Credit и в таблице Orders и т.д. После выявления всех соответствующих записей опция выполняет ту же процедуру для второй записи в таблице Customer.

Customer Table Customer Name	Credit Table Amount	Orders Table Order Amount
Cyclists Incorporated	(\$1088.56)	\$1529.70
Cyclists Incorporated	(\$1260.12)	\$23.50
CyclePath Corp.	(\$1958.03)	\$49.50
CyclePath Corp.	(\$1076.43)	\$1702.60
CyclePath Corp.	(\$75.04)	
The Great Bike Shop	(\$138.98)	\$3269.70
The Great Bike Shop		\$5219.55
The Great Bike Shop		\$1538.20

Помните, что для каждой строки в отчете значение в поле Amount таблицы Credit не обязательно должно иметь связь со значением поля Order Amount таблицы Orders. Однако, для каждой записи в таблице Customer одновременно выбирается одна запись в таблице Credit и одна запись в таблице Orders.

Помните также, что когда в таблице Credit заканчиваются записи, соответствующие данной записи в таблице Customer, вместо значений будут отображаться пробелы, до тех пор, пока соответствующие значения будут находиться в таблице Orders, и наоборот.

Поиск всех соответствий в одной, а затем в другой таблице (A – B, A - C)

Для каждой записи в таблице Customer данная опция ищет все соответствующие записи сначала в таблице Credit (таблица B), а затем в таблице Orders (таблица C). После выявления всех соответствующих записей процедура повторяется для второй записи в таблице Customer.

<i>Customer Table</i> Customer Name	<i>Credit Table</i> Amount	<i>Orders Table</i> Order Amount
Cyclists Incorporated	(\$1088.56)	
Cyclists Incorporated	(\$1260.12)	
Cyclists Incorporated		\$1529.70
Cyclists Incorporated		\$23.50
CyclePath Corp.	(\$1958.03)	
CyclePath Corp.	(\$1076.43)	
CyclePath Corp.	(\$75.04)	
CyclePath Corp.		\$49.50
CyclePath Corp.		\$1702.60
The Great Bike Shop	(\$138.98)	
The Great Bike Shop		\$3269.70
The Great Bike Shop		\$5219.55
The Great Bike Shop		\$1538.20

ВНИМАНИЕ: Если Вы хотите, чтобы данные Таблицы C (в данном примере - таблица Orders) располагались в Вашем отчете перед данными Таблицы B (таблица Credit), то Вам придется изменить связь так, чтобы первой шла связь A – C, а затем A – B. Это можно сделать с помощью Visual Linking Expert. Для изменения порядка связей удалите существующие связи, а затем установите их снова, но в требуемом порядке.

Поиск всех комбинаций двух файлов

Для каждой записи в таблице Customer данная опция ищет первую соответствующую запись в таблице Credit, а затем все соответствующие записи в таблице Orders. После выявления в таблице Orders всех соответствующих записей опция ищет в этой же таблице все записи, соответствующие второй записи в таблице Credit, затем следующей записи и т.д. После того, как в таблице Orders выявлены все соответствия для всех записей из таблицы Credit, которые

соответствуют первой записи в таблице Customer, процедура повторяется для второй записи в таблице Customer.

Customer Table Customer Name	Credit Table Amount	Orders Table Order Amount
Cyclists Incorporated	(\$1088.56)	\$1529.70
Cyclists Incorporated	(\$1088.56)	\$23.50
Cyclists Incorporated	(\$1260.12)	\$1529.70
Cyclists Incorporated	(\$1260.12)	\$23.50
CyclePath Corp.	(\$1958.03)	\$49.50
CyclePath Corp.	(\$1958.03)	\$1702.60
CyclePath Corp.	(\$1076.43)	\$49.50
CyclePath Corp.	(\$1076.43)	\$1702.60
CyclePath Corp.	(\$75.04)	\$49.50
CyclePath Corp.	(\$75.04)	\$1702.60
The Great Bike Shop	(\$138.98)	\$3269.70
The Great Bike Shop	(\$138.98)	\$5219.55
The Great Bike Shop	(\$138.98)	\$1538.20

ВНИМАНИЕ: Если Вы хотите, чтобы программа искала сначала соответствующую запись в Таблице С (в данном примере - таблица Orders), а затем все соответствующие записи в таблице В (таблица Credit), т.е. выполняла процесс, обратный описанному выше, то Вам придется изменить связи так, чтобы первой шла связь А – С, а затем А – В. Это можно сделать с помощью Visual Linking Expert. Для изменения порядка связей, удалите существующие связи, а затем установите их снова, но в требуемом порядке.

Типы соединения SQL (источники данных ODBC)

Crystal Report Designer позволяет Вам задавать тип соединения, который Вы хотите использовать при установлении связи между таблицами SQL. Соединение SQL показывает, как сопоставляются связанные поля в двух таблицах SQL при считывании записей. Опции SQL Join Type задаются с помощью диалогового окна Link Options.

ВНИМАНИЕ: При установлении связей с помощью соединений SQL индексированные поля не требуются.

Существуют следующие типы соединений:

- Соединение Equal [=]
- Соединение Left Outer [= (+), * =]
- Соединение Right Outer [(+) =, =*]
- Соединение Greater Than [>]
- Соединение Less Than [<]
- Соединение Greater Than Or Equal [>=]
- Соединение Less Than Or Equal [<=]
- Соединение Not Equal [!=, <>]

Соединение Equal [=]

Набор, получаемый в результате соединения Equal, включает в себя все записи, в которых значения связанных полей в обеих таблицах полностью соответствуют. В следующем примере таблица Customer связана с таблицей Orders по полю Customer ID. Когда Customer ID, обнаруженный в таблице Orders, соответствует Customer ID в таблице Customers, в обеих таблицах отображается информация из соответствующих записей.

Для описания соединения Equal SQL использует следующий синтаксис:

```
SELECT Customer.'Customer ID' ,
       Customer.'Customer Name' ,
       Orders.'Order Amount'
FROM 'Customer' Customer ,
     'Orders' Orders
WHERE Customer.Customer ID =
       Orders.Customer ID
```

Данный оператор выводит следующие данные:

<i>Customer Table Customer ID</i>	<i>Customer Table Customer Name</i>	<i>Orders Table Order Amount</i>
52	Allez Distribution	25141.50
53	BG Mountain Inc.	19164.30
53	BG Mountain Inc.	1683.60
57	Hansen MTB Inc.	15716.40
58	La Bomba de Bicicleta	1956.20
60	Mountain Toad	24580.50
62	SFB Inc.	7911.80
63	Sierra Bicycle Group	19766.20
63	Sierra Bicycle Group	12763.95
64	Sierra Mountain	8233.50

Соединение Left Outer [= (+), * =]

Набор, получаемый в результате соединения Left Outer, включает в себя все записи, в которых значения связанного поля в обеих таблицах полностью соответствуют. В него также входит строка для каждой записи в первичной (левой) таблице, значение связанного поля которой не имеет соответствия в поисковой таблице. Например, Вы можете использовать соединение Left Outer для просмотра всех заказчиков и размещенных ими заказов, но при просмотре Вы также получите строки с именами заказчиков, которые не делали заказы. Такие заказчики будут отображены в конце списка с пробелами в поле, которое должно содержать информацию о заказе.

<i>Customer Table</i> <i>Customer ID</i>	<i>Customer Table</i> <i>Customer Name</i>	<i>Orders Table</i> <i>Order Amount</i>
52	Allez Distribution	25141.50
53	BG Mountain Inc.	19164.30
53	BG Mountain Inc.	1683.60
57	Hansen MTB Inc.	15716.40
58	La Bomba de Bicicleta	1956.20
60	Mountain Toad	24580.50
62	SFB Inc.	7911.80
63	Sierra Bicycle Group	19766.20
63	Sierra Bicycle Group	12763.95
64	Sierra Mountain	8233.50
54	Bicicletas Aztecas	
55	Deely MTB Inc.	

ВНИМАНИЕ: Соединения *Left Outer* и *Right Outer* обрабатываются на языке SQL не так, как все остальные типы соединений. Если доступ к базе данных осуществляется посредством ODBC, то Crystal Report Designer будет использовать в операторе SQL синтаксис ODBC. Если Вы присоединяетесь к базе данных SQL непосредственно (не через ODBC), то Crystal Report Designer будет использовать синтаксис этой базы данных. Для получения более полной информации о виде соединения *Outer* в операторе SQL обратитесь к документации по ODBC или к документации для Вашей базы данных SQL.

Соединение **Right Outer [(+) =, =*]**

Набор, получаемый в результате соединения *Right Outer*, включает в себя все записи, в которых значения связанного поля в обеих таблицах полностью соответствуют. В него также входит строка для каждой записи в поисковой (правой) таблице, значение связанного поля которой не имеет соответствия в первичной таблице. Если Вы связываете таблицу *Customer* с таблицей *Orders*, то в таблице будет присутствовать строка для каждого заказа, размещенного заказчиком, как это происходит при соединении *Equal*. Кроме того, будет присутствовать строка и для тех заказов, которые оказались невозможным связать с заказчиком. Теоретически, такое не должно произойти, но если неопытный менеджер по продажам забудет присвоить *Customer ID* заказу, Вы быстро сможете выявить такой заказ при помощи соединения *Right Outer*. В полученной таблице для таких заказов без заказчиков поле *Customer* будет пустым.

Customer Table Customer ID	Orders Table Order ID	Orders Table Order Amount
52	6	25141.50
53	11	19164.30
53	21	1683.60
57	4	15716.40
58	20	1956.20
60	16	24580.50
62	19	7911.80
63	28	19766.20
63	32	12763.95
64	14	8233.50
	25	10320.87

ВНИМАНИЕ: Соединения *Left Outer* и *Right Outer* обрабатываются на языке SQL не так, как все остальные типы соединений. Если доступ к базе данных осуществляется посредством ODBC, то Crystal Report Designer будет использовать в операторе SQL синтаксис ODBC. Если Вы присоединяетесь к базе данных SQL непосредственно (не через ODBC), то Crystal Report Designer будет использовать синтаксис этой базы данных. Для получения более полной информации о виде соединения *Outer* в операторе SQL обратитесь к документации по ODBC или к документации для Вашей базы данных SQL.

Соединение Greater Than [>]

Набор, получаемый в результате соединения Greater Than, включает в себя все записи, в которых значение связанного поля из первичной таблицы больше значения связанного поля из поисковой таблицы. Например, компании может потребоваться сравнить продажи всех ее торговых представителей с продажами всех менеджеров по продажам. Руководство компании желает удостовериться в том, что каждый менеджер приносит больше прибыли, чем любой представитель. Имея это в виду, с помощью соединения Greater Than Вы можете связать таблицу SalesRep с таблицей Manager по полю Salary, присутствующему в обеих таблицах:

```
SELECT SalesRep.'Last Name' ,
       SalesRep.'Salary' ,
       Manager.'Last Name' ,
       Manager.'Salary'
FROM 'SalesRep' SalesRep ,
     'Manager' Manager
WHERE SalesRep.'Salary' >
      Manager.'Salary'
```

Оператор SQL может вывести данные, аналогичные следующим:

<i>SalesRep Table</i> <i>Last Name</i>	<i>SalesRep Table</i> <i>Salary</i>	<i>Manager Table</i> <i>Last Name</i>	<i>Manager Table</i> <i>Salary</i>
Davolio	\$35,000.00	Fuller	\$32,000.00
Davolio	\$35,000.00	Brid	\$30,000.00
Davolio	\$35,000.00	Buchanan	\$29,500.00
Dodsworth	\$48,300.00	Hellstern	\$45,000.00
Dodsworth	\$48,300.00	Fuller	\$32,000.00
Dodsworth	\$48,300.00	Brid	\$30,000.00
Dodsworth	\$48,300.00	Buchanan	\$29,500.00
Dodsworth	\$48,300.00	Martin	\$35,000.00
Patterson	\$30,000.00	Buchanan	\$29,500.00

В данной таблице отношения между торговыми представителями и торговыми менеджерами не установлены. Так как все менеджеры считаются выше по должности, чем торговые представители, руководство компании сравнивает различные аспекты продаж, произведенных торговыми менеджерами и представителями, для выявления несоответствий в должности.

Соединение Less Than [<]

Набор, получаемый в результате соединения Less Than, включает в себя все записи, в которых значение связанного поля из первичной таблицы меньше значения связанного поля из поисковой таблицы. Используя соединение Less Than, Вы можете сравнить различные аспекты продаж, произведенных торговыми менеджерами и торговыми представителями. И снова, поле Salary в каждой таблице используется как поле связи. Однако, в этом случае с помощью соединения Less Than связь устанавливается от таблицы Manager к таблице SalesRep по связанным полям Salary:

```
SELECT Manager.'Last Name' ,
       Manager.'Salary' ,
       SalesRep.'Last Name' ,
       SalesRep.'Salary'
FROM 'Manager' Manager ,
     'SalesRep' SalesRep
WHERE Manager.'Salary' <
       SalesRep.'Salary'
```

Таблица, выводимая оператором SQL в данном случае, несколько отличается от таблицы для соединения Greater Than:

<i>Manager Table</i> <i>Last Name</i>	<i>Manager Table</i> <i>Salary</i>	<i>SalesRep Table</i> <i>Last Name</i>	<i>SalesRep Table</i> <i>Salary</i>
Fuller	\$32,000.00	Davolio	\$35,000.00
Fuller	\$32,000.00	Dodsworth	\$48,300.00
Brid	\$30,000.00	Davolio	\$35,000.00
Brid	\$30,000.00	Dodsworth	\$48,300.00
Buchanan	\$29,500.00	Davolio	\$35,000.00
Buchanan	\$29,500.00	Dodsworth	\$48,300.00
Buchanan	\$29,500.00	Patterson	\$30,000.00
Martin	\$35,000.00	Dodsworth	\$48,300.00
Hellstern	\$45,000.00	Dodsworth	\$48,300.00

Соединение Greater Than Or Equal [>=]

Набор, получаемый в результате соединения Greater Than Or Equal, включает в себя все записи, в которых значение связанного поля из первичной таблицы больше или равно значению связанного поля из поисковой таблицы. Пример, приведенный для данного соединения, аналогичен примеру для соединения Greater Than, только в нем использовано соединение Greater Than Or Equal:

```
SELECT SalesRep.'Last Name' ,
       SalesRep.'Salary' ,
       Manager.'Last Name' ,
       Manager.'Salary'
FROM 'SalesRep' SalesRep,
     'Manager' Manager
WHERE SalesRep.'Salary' >=
      Manager.'Salary'
```

Оператор SQL может вывести данные, аналогичные следующим:

<i>SalesRep Table</i> <i>Last Name</i>	<i>SalesRep Table</i> <i>Salary</i>	<i>Manager Table</i> <i>Last Name</i>	<i>Manager Table</i> <i>Salary</i>
Davolio	\$35,000.00	Fuller	\$32,000.00
Davolio	\$35,000.00	Brid	\$30,000.00
Davolio	\$35,000.00	Buchanan	\$29,500.00
Davolio	\$35,000.00	Martin	\$35,000.00
Dodsworth	\$48,300.00	Hellstern	\$45,000.00
Dodsworth	\$48,300.00	Fuller	\$32,000.00
Dodsworth	\$48,300.00	Brid	\$30,000.00
Dodsworth	\$48,300.00	Buchanan	\$29,500.00
Dodsworth	\$48,300.00	Martin	\$35,000.00
Patterson	\$30,000.00	Brid	\$30,000.00
Patterson	\$30,000.00	Buchanan	\$29,500.00

Соединение Less Than Or Equal [<=]

Набор, получаемый в результате соединения Less Than Or Equal, включает в себя все записи, в которых значение связанного поля из первичной таблицы меньше или равно значению связанного поля из поисковой таблицы. Пример, приведенный для данного соединения, аналогичен примеру для соединения Less Than, только в нем использовано соединение Less Than Or Equal:

```
SELECT Manager.'Last Name',
       Manager.'Salary',
       SalesRep.'Last Name',
       SalesRep.'Salary'
FROM 'Manager' Manager,
     'SalesRep' SalesRep
WHERE Manager.'Salary' <=
       SalesRep.'Salary'
```

Оператор SQL может вывести данные, аналогичные следующим:

<i>Manager Table</i>	<i>Manager Table</i>	<i>SalesRep Table</i>	<i>SalesRep Table</i>
<i>Last Name</i>	<i>Salary</i>	<i>Last Name</i>	<i>Salary</i>
Fuller	\$32,000.00	Davolio	\$35,000.00
Fuller	\$32,000.00	Dodsworth	\$48,300.00
Brid	\$30,000.00	Davolio	\$35,000.00
Brid	\$30,000.00	Dodsworth	\$48,300.00
Brid	\$30,000.00	Patterson	\$30,000.00
Buchanan	\$29,500.00	Davolio	\$35,000.00
Buchanan	\$29,500.00	Dodsworth	\$48,300.00
Buchanan	\$29,500.00	Patterson	\$30,000.00
Martin	\$35,000.00	Davolio	\$35,000.00
Martin	\$35,000.00	Dodsworth	\$48,300.00
Hellstern	\$45,000.00	Dodsworth	\$48,300.00

Соединение Not Equal [!=, <>]

Набор, получаемый в результате соединения Not Equal, включает в себя все записи, в которых значение связанного поля из первичной таблицы не равно значению связанного поля из поисковой таблицы. Данный тип соединения может использоваться для поиска возможных комбинаций элементов в случаях, когда таблица соединена сама с собой (само-соединение). Например, компания может вести таблицы, включающие в себя все продаваемые ею продукты. Если в компании принимается решение о ведении продаж, когда заказчики, покупающие один продукт, смогут купить второй со скидкой, то, возможно, потребуется создать список всех возможных комбинаций из двух продуктов:

```
SELECT Product1.'Product Name' ,
       Product2.'Product Name' ,
FROM 'Product' Product1
     'Product' Product2
WHERE Product1.'Product Name' !=
       Product2.'Product Name'
```

В данном операторе SQL таблица Product открывается дважды. Первый раз для нее задается имя псевдонима Product1, а во второй раз – имя псевдонима Product2. После этого, для установления связи от таблицы Product1 к таблице Product2 используется поле Product Name. Это одна и та же таблица, но, т.к. она была открыта дважды с использованием различных псевдонимов, Crystal Report Designer будет рассматривать ее как две отдельные таблицы. Для установки связи между таблицами по полю Product Name используется соединение Not Equal. В результате, каждый продукт комбинируется с каждым другим предлагаемым продуктом, но не с самим собой.

Product 1	Product 2
Product Name	Product Name
Craze Adult Helmet	Craze Mtn Lock
Craze Adult Helmet	InFlux Lucra Glove
Craze Adult Helmet	Roadster Micro Mtn Saddle
Craze Mtn Lock	Craze Mtn Lock
Craze Mtn Lock	InFlux Lucra Glove
Craze Mtn Lock	Roadster Micro Mtn Saddle
InFlux Lucra Glove	Craze Adult Helmet
InFlux Lucra Glove	Craze Mtn Lock
InFlux Lucra Glove	Roadster Micro Mtn Saddle
Roadster Micro Mtn Saddle	Craze Adult Helmet
Roadster Micro Mtn Saddle	Craze Mtn Lock
Roadster Micro Mtn Saddle	InFlux Lucra Glove

ВНИМАНИЕ: Символ != используется для представления соединения *Not Equal*, если драйвер источника данных ODBC для данных, к которым осуществляется доступ, поддерживает этот символ. Если нет, то для представления соединения *Not Equal* используется принимаемый по умолчанию символ <>. Смотрите *Self join* в оперативной Подсказке *Crystal Report Designer*.

Использование SQL и баз данных SQL

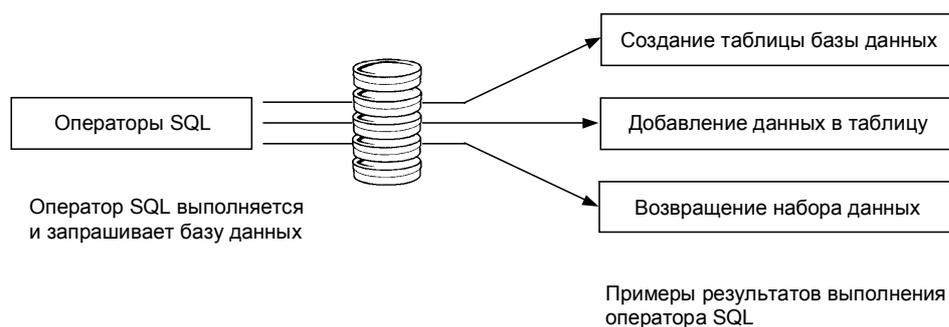
Наиболее популярными и мощными форматами баз данных, возможно, являются приложения DBMS на основе Языка структурированных запросов (SQL). Базы данных SQL обычно работают в сетевой архитектуре клиент/сервер, предоставляя Сервер SQL для создания, сохранения, манипуляции файлами, таблицами, полями и записями базы данных и интерфейс Клиента SQL, чтобы пользователи рабочей станции могли не только разрабатывать и работать с файлами базы данных, но также искать полезные значимые данные, помогающие им в ежедневной работе.

Что такое SQL?

Язык запросов SQL разработан для осуществления организации, управления, разработки и запросов в больших реляционных базах данных через компьютерную сеть. Язык SQL принят в сфере Информационной науки (IS) и Информационного управления. Язык был стандартизован Американским национальным институтом стандартизации (ANSI) и Международной организацией стандартизации (ISO). Это означает, что в любой версии SQL, созданной компанией по разработке программного обеспечения и заявленной как версия SQL, должны присутствовать определенные свойства. Многие разработчики программного обеспечения добавляют в свои версии SQL дополнительные расширенные свойства с целью усовершенствования языка и привлечения покупателей, но, при этом, такая версия должна нести в себе оригинальные стандарты, установленные ANSI и ISO.

Необходимо понимать, что SQL не является истинно компьютерным языком. Он не может использоваться для создания автономных компьютерных приложений или операционных систем. SQL часто называют под-языком, который может использоваться из других языков и приложений. Что наиболее важно, так это то, что язык SQL специально предназначен для работы с реляционными базами данных.

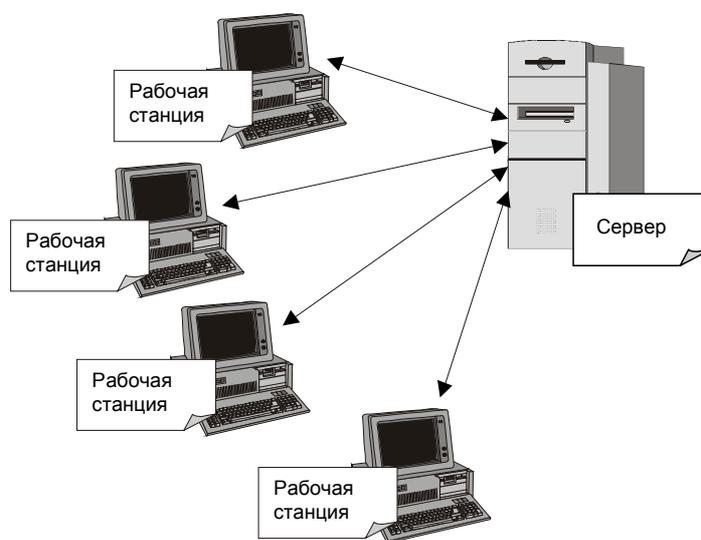
Синтаксис языка SQL строится на системе посылки операторов SQL на сервер базы данных SQL. Каждый оператор представляет собой запрос о выполнении операции с базой данных, такой как создание файла базы данных, добавление таблиц и полей в базу данных, добавление записей в таблицы или поиск данных из баз данных. Сервер SQL анализирует оператор SQL и выполняет требуемую операцию. Если оператор представляет запрос о данных, то сервер собирает данные и возвращает их на клиентскую рабочую станцию для просмотра пользователями.



Запрос SQL является оператором SQL, созданным специально для запрашивания данных из одной или нескольких баз данных. Некоторые приложения SQL требуют от Вас прямого ввода запроса SQL с помощью текстового редактора, другие поддерживают графический пользовательский интерфейс, проводящий Вас через процесс запроса базы данных SQL. В последнем случае, приложение должно создавать оператор SQL на основе предоставляемой Вами информации. Такой оператор является фактическим запросом SQL, т.е. именно тем запросом, который используется для запрашивания данных. Crystal Report Designer подпадает под эту последнюю категорию SQL-совместимых приложений.

Архитектура клиент/сервер

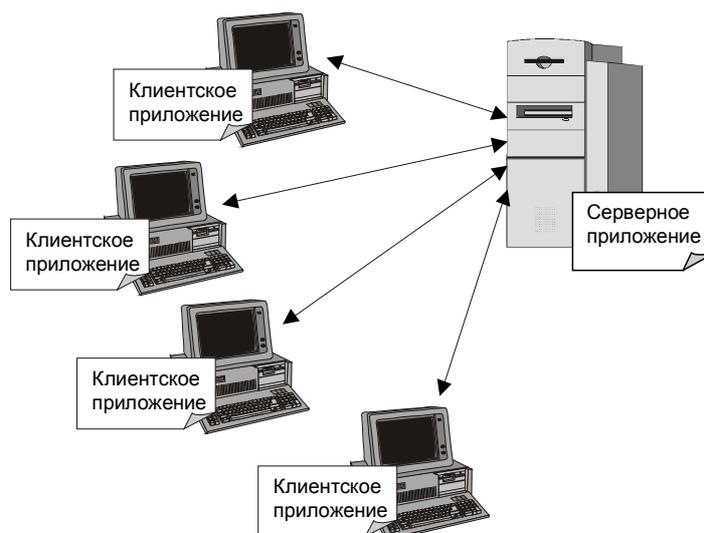
Одним из наиболее важных свойств приложений SQL DBMS является их возможность эффективно использовать сетевую архитектуру клиент/сервер.



Простая сеть состоит из одного или нескольких серверов, указывающих общее расположение, чтобы все пользователи в сети могли получать данные и приложения. Кроме того, многие сетевые серверы обеспечивают сетевую защиту, автоматизированные службы, такие как резервирование данных, и мониторинг сетевых ресурсов для поддержания наилучшего качества обслуживания всех рабочих станций в сети. В связи с высокими требованиями по обработке, предъявляемыми сетевому серверу, в качестве сервера часто используется мощный компьютер с высоким быстродействием, который может включать в себя несколько процессоров, несколько дисководов для жестких дисков и несколько дисководов для CD-ROM.

Клиентом сети является рабочая станция с одним компьютером, которая регулярно используется одним или несколькими работниками компании. Пользователь работает на клиенте и имеет доступ к данным и приложениям сервера по сети. Большие задания по обработке, требующие больших временных и ресурсных затрат, обрабатываются сервером, а конечные результаты посылаются клиенту. Это помогает пользователям сберечь время, т.к. на локальной рабочей станции меньше времени используется на обработку и остается больше времени пользователями.

На такой архитектуре клиент/сервер основаны многие современные компьютерные приложения. Простое приложение клиент/сервер состоит из двух частей: серверного приложения, размещенного на сетевом сервере, и клиентского приложения, размещенного на рабочей станции пользователя. Приложение сервера обрабатывает сложные, требующие больших временных затрат или большой мощности процессы, используя для этого компьютерные ресурсы сетевого сервера, а клиентское приложение поддерживает простой пользовательский интерфейс, помогающий улучшить и ускорить работу.



В зависимости от стоимости, приложение клиент/сервер часто обеспечивает определенное количество мест. Каждое место представляет собой одну клиентскую рабочую станцию или, по выбору создателя программного обеспечения, одного пользователя, которые могут присоединяться и использовать программное обеспечение клиент/сервер. Часто производители программного обеспечения продают дополнительные места для своих приложений. Каждое такое место поставляется с полным набором программного обеспечения клиентского приложения.

Не следует путать серверные приложения с компьютером сетевого сервера. Оба они часто называются серверами. Однако серверное приложение размещается на сетевом сервере, используя все преимущества аппаратного обеспечения и операционной системы машины-сервера. Сетевой сервер представляет собой физическую машину, к которой клиенты сети присоединены с помощью кабелей или других соединительных устройств.

SQL DBMS

Система управления базами данных SQL является наиболее общим примером программного пакета клиент/сервер. В стандартную SQL DBMS входит приложение сервера SQL, выполняющее всю фактическую работу по построению и работе с базами данных и находящимися в них данными. Кроме того, DBMS включает в себя, как минимум, один набор клиентского программного обеспечения SQL (одно место), которое может быть соединено с сервером по сети. Клиентское программное обеспечение SQL состоит, как минимум, из редактора операторов SQL, который может использоваться для написания и выполнения операторов SQL, и базового коммуникационного уровня, работающего с серверным приложением SQL по сети.

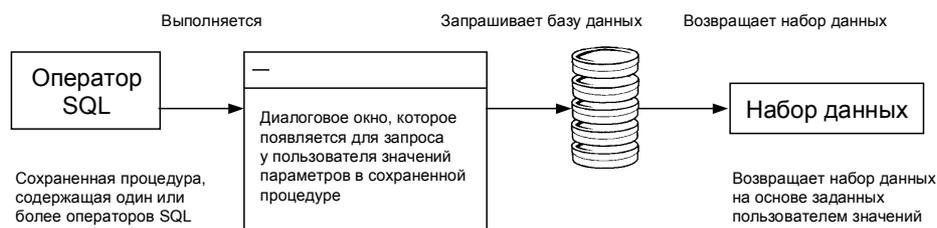
При выполнении Вами оператора SQL клиентское программное обеспечение передает этот оператор на коммуникационный уровень, который посылает оператор по сети на сервер. Сервер SQL анализирует оператор, выполняет требуемую операцию и возвращает запрошенные данные клиенту. Если клиентская программа получает от сервера какие-либо данные, она отображает их для пользователя.

Клиент SQL, осуществляющий связь с сервером SQL



Сохраненные процедуры

Дополнительно к обычным атрибутам, таблицам, полям и записям реляционной базы данных многие системы SQL DBMS поддерживают сохраненные процедуры. Сохраненная процедура представляет собой скомпилированную программу SQL, состоящую из одного или нескольких операторов. Сохраненная процедура может применяться для определения запроса SQL, который затем может использоваться множество раз. Более того, в сохраненных процедурах могут определяться переменные, условные выражения и аргументы переменных, обеспечивая возможность запроса у пользователя информации перед выполнением процедуры.



Так как сохраненные процедуры могут возвращать набор результатов, то они также могут поддерживать определенный набор данных при выполнении. Фактически, Crystal Report Designer позволяет выполнять сохраненные процедуры на базе данных SQL и использовать возвращенные данные при создании отчета. Если сохраненная процедура включает в себя запрашивание у пользователя информации, на которой она будет основывать свой запрос, Crystal Report Designer будет запрашивать эту информацию при выборе Вами такой сохраненной процедуры для своего отчета. Смотрите *Как выбирать сохраненную процедуру из базы данных SQL и изменять параметры сохраненной процедуры*.

Как Crystal Report Designer использует SQL?

Если Вы присоединяетесь к базе данных SQL или любой базе данных ODBC, Crystal Report Designer будет действовать как клиентское приложение SQL, соединяя Вас с сервером по сети.

Если Вы создаете отчет с доступом к данным SQL, Crystal Report Designer строит запрос SQL. Если Вы знаете SQL и чувствуете, что созданный запрос можно улучшить, Вы можете это сделать. При выборе Вами команды **Show SQL Query** из меню Database, в

диалоговом окне Show SQL Query отобразится запрос SQL, созданный Crystal Report Designer.

Запрос SQL является представлением оператора SQL, который Crystal Report Designer посылает на сервер. Переводя максимально возможную часть Вашего отчета в запрос SQL, Crystal Report Designer может переложить большую часть обработки отчета на сервер. Вместо того, чтобы самому просматривать всю базу данных в поиске запрошенных Вами данных, Crystal Report Designer передаст эту работу серверу и получит от него значительно меньший объем данных, сохраняя таким образом время и ресурсы Вашей рабочей станции.

Язык SQL

Так как для получения доступа к базам данных клиента/сервер через ODBC Crystal Report Designer использует язык SQL, Вам будет легче разобраться в процессе генерации отчета, если Вы поймете некоторые используемые выражения (команды) SQL:

SELECT

Выражение SELECT указывает определенные элементы данных, которые нужно найти в таблицах базы данных. Возвращаемый элемент может быть представлен значениями в поле (колонке) базы данных или же результатом вычисления, выполненного во время сбора данных. Например:

```
SELECT
    TABLEA. 'CUSTNAME' ,
    TABLEA. 'STATE '
```

DISTINCT

Выражение DISTINCT заставляет запрос искать только уникальные наборы данных. Ряд результатов будет возвращен только один раз. Приведенный выше оператор Select можно изменить, чтобы в нем использовалось выражение DISTINCT:

```
SELECT DISTINCT
    TABLEA. 'CUSTNAME' ,
    TABLEA. 'STATE '
```

FROM

Выражение FROM задает источники полей базы данных, указанных в выражении SELECT. FROM перечисляет фактические таблицы базы данных, содержащие поля и записи, в которых присутствуют запрошенные данные. Выражение FROM, генерированное Crystal Report Designer, предваряет имя каждой таблицы псевдонимом, используемым для идентификации таблицы в Вашем отчете. Ниже представлен пример выражения FROM в сочетании с выражением SELECT:

```
SELECT
    TABLEA. 'CUSTNAME' ,
    TABLEA. 'STATE '

FROM
    'TABLEA' TABLEA
```

WHERE

Выражение WHERE имеет два назначения:

1. WHERE может задавать критерии выбора записи.
2. WHERE может задавать, как объединены две таблицы базы данных.

Если WHERE используется для задания критериев выбора записи, то будет возвращено условие поиска, определяющее, какие записи (ряды данных) должны быть найдены. Например:

```
SELECT
    MYTABLE . ' SALESPERSON ' ,
    MYTABLE . ' SALESTOTAL '
FROM
    'MYTABLE' MYTABLE
WHERE
    MYTABLE . ' SALESTOTAL ' < 10000.00
```

Если WHERE задает, как связаны две таблицы, между двумя именами таблиц находится оператор соединения SQL. Смотрите *Типы соединений SQL (источники данных ODBC)*. Ниже приведен пример, когда выражение WHERE соединяет две таблицы:

```
SELECT
    CUSTOMER . ' CUST_ID ' ,
    CUSTOMER . ' CUST_NAME ' ,
    ORDERS . ' AMOUNT '
FROM
    'CUSTOMER' CUSTOMER ,
    'ORDERS' ORDERS
WHERE
    CUSTOMER . ' CUST_ID ' = ORDERS . ' CUST_ID '
```

ORDER BY

Выражение ORDER BY указывает, что возвращенные записи базы данных должны быть отсортированы по значениям в определенном поле. Если выражение ORDER BY не используется, то записи будут возвращены в том порядке, в котором она располагаются в исходной базе данных. Если после выражения ORDER BY задано более одного поля, то данные будут отсортированы по значениям в первом заданном поле, затем, в пределах этой сортировки, по значениям во втором поле и т.д. В приведенном ниже операторе SQL использовано выражение ORDER BY:

```
SELECT
    MYTABLE . ' COMPANY ' ,
    MYTABLE . ' CITY ' ,
    MYTABLE . ' STATE '
FROM
    'MYTABLE' MYTABLE
ORDER BY
    MYTABLE . ' STATE ' ASC ,
    MYTABLE . ' CITY ' ASC
```

ВНИМАНИЕ: ASC показывает, что значения в поле отсортированы в возрастающем, а не в убывающем (DESC) порядке. В возрастающем порядке сортируются буквы от А до Z и числа от 0 до 9.

GROUP BY

Выражение GROUP BY возвращает набор суммированных данных. Вместо поиска самих данных, GROUP BY группирует данные и суммирует каждую группу в соответствии с функцией суммирования SQL. В Crystal Report Designer возвращается только суммированная информация для каждой группы. Например:

```
SELECT
    MYTABLE . ' STATE ' ,
    MYTABLE . ' ZIPCODE ' ,
    SUM ( MYTABLE . ' SALES ' )
FROM
    ' MYTABLE ' MYTABLE
GROUP BY
    MYTABLE . ' STATE ' ,
    MYTABLE . ' ZIPCODE '
```

HAVING

Выражение HAVING создает критерии выбора для суммированной информации, выдаваемой выражением GROUP BY. Функция выражения HAVING аналогична функции выбора записей выражения WHERE, но HAVING применяется к суммированным данным, создаваемым выражением GROUP BY. Поэтому в любом операторе SQL Crystal Report Designer, создающем суммированные результаты, перед выражением HAVING всегда будет находиться выражение GROUP BY. Это видно из следующего примера:

```
SELECT
    TABLE . ' CUSTOMER ' ,
    SUM ( TABLE . ' ORDERAMOUNT ' )
FROM
    ' TABLE ' TABLE
GROUP BY
    TABLE . ' CUSTOMER '
HAVING
    SUM ( TABLE . ' ORDERAMOUNT ' ) > 20000.00
```

Дополнительная информация

В настоящей главе затронуты только некоторые из наиболее важных аспектов, связанных с доступом к базе данных, реляционными базами и SQL. Если Вы хотите более детально изучить вопросы, касающиеся баз данных, обратитесь к документации к Вашему приложению DBMS.

Кроме того, существует много различной литературы, во всей полноте описывающей теорию баз данных и их разработку.

ПРАКТИКА (Работа с базами данных)

В настоящем разделе приведены пошаговые инструкции по выполнению некоторых общих процедур, связанных с доступом к файлам базы данных из Crystal Report Designer. Используйте эти процедуры для ускорения работы с программой.

Как открывать запросы Access через DAO

Запросы Access Microsoft могут использоваться в Crystal Report Designer как отдельные наборы данных, подобно таблицам Access. При открытии базы данных Access посредством механизма DAO (см. *Microsoft Access*) любой запрос в базе данных может считываться автоматически.

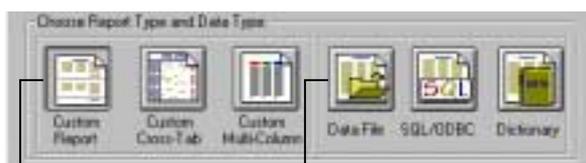
ВНИМАНИЕ: Механизм DAO присутствует только в 32-битовой версии Crystal Report Designer. Информация, приведенная в настоящем разделе, предполагает использование 32-разрядного механизма DAO.

1. Щелкните мышкой по кнопке **New** на стандартной панели инструментов.



- 2 Щелкните кнопку Custom

Report Gallery (Галерея отчета) расширится.



- 3 Щелкните кнопку Custom Report
- 4 Щелкните кнопку Data File

На экране отобразится диалоговое окно Choose Database File.

5. Найдите и выделите файл базы данных Microsoft Access (*.MDB), содержащий запрос, который Вы хотите использовать в своем отчете.
6. Нажмите **OK**. В окне приложения отобразится Design Tab с диалоговым окном Insert Fields.
7. Все таблицы и запросы из Вашей базы данных Access появятся в окне списка в Database Tab диалогового окна Insert Fields. Запросы в списке будут идти после таблицы. Найдите свой запрос в Database Tab и дважды щелкните мышкой по имени запроса в списке. Запрос расширится, отобразив все имеющиеся в нем поля.

8. Выберите те поля, которые Вы хотите использовать в своем отчете и добавьте их в Design Tab.

ВНИМАНИЕ: В Crystal Report Designer нельзя использовать запросы действия или обновления Access. Вы можете использовать только запросы выбора и комбинационных таблиц.

Как открывать запросы Access через ODBC

ODBC предоставляет Вам больше возможностей для контроля за теми частями базы данных, которые Вы хотите использовать. В связи с этим, для использования запроса Access через ODBC Вам потребуется выполнить некоторые дополнительные шаги:

1. Выберите команду **Options** из меню File. На экране отобразится диалоговое окно File Options. Щелкните мышкой по SQL Tab, чтобы отобразились опции SQL и ODBC.



- 2 Удостоверьтесь, что поставлены флажки Views или Prompt on every table

- Опция *Views* автоматически отображает все доступные запросы в вашей базе данных Access.
 - Опция *Prompt on every table* будет отображать диалоговое окно Allow Reporting On каждый раз, когда Вы будете присоединяться к источнику данных ODBC. Это диалоговое окно поддерживает те же опции, что и раздел *Allow Reporting On* в диалоговом окне File Options, но эти опции будут поддерживаться для каждого источника данных ODBC, к которому Вы присоединитесь.
3. Кроме того, если хотите, Вы можете задать опции *Table Name LIKE* и *Owner LIKE*.
 - *Table Name LIKE* основана на выражении SQL LIKE. Эта опция позволяет Вам задавать виды имен таблиц, которые должны отображаться в диалоговом окне Choose SQL Table. В качестве специальных символов используйте с данной функцией символ подчеркивания () или знак процента (%). Символ подчеркивания задает один символ, а знак процента – строку символов. Например, если Вы зададите DAV_, то отобразится только DAV, а если зададите DAV%, отобразятся DAVE и

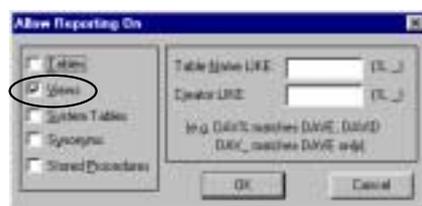
DAVID. *Table Name LIKE C%* будет отображать только те таблицы, имена которых начинаются с буквы С.

- *Owner LIKE* также основана на выражении SQL LIKE. Опция *Owner LIKE* позволяет Вам выбрать Owner (Владельца) (или Creator (Создателя) или Alias (Псевдоним)) таблицы, а не само имя таблицы. Например, *Owner LIKE C%* отобразит только те таблицы, имя владельца которых начинается в буквы С.
4. Нажмите *OK* для выхода из диалогового окна File Options.
 5. Создайте новый отчет. При выборе Вами SQL/ODBC в качестве источника данных на экране отобразится диалоговое окно Log On Server.



6 Выберите источник данных ODBC для своей базы данных ACCESS, затем нажмите OK.

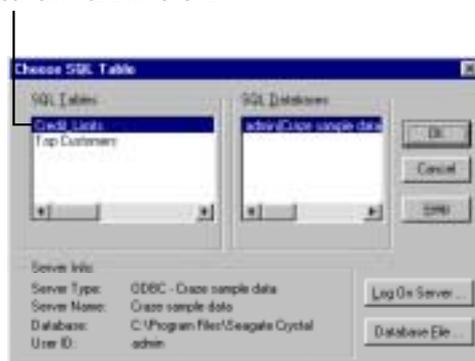
7. Если Вы не задали определенный файл базы данных Access в источнике данных ODBC Access, то на экране отобразится диалоговое окно Select Database. Найдите и выберите базу данных, содержащую запрос Access, который Вы хотите использовать, затем нажмите *OK*.
8. Если в диалоговом окне File Options Вы выбрали флажок *Prompt on every table*, то отобразится диалоговое окно Allow Reporting On. В остальных случаях перейдите к Шагу 10.



9 Удостоверьтесь, что флажок Views установлен, затем нажмите OK.

На экране отобразится диалоговое окно Choose SQL Table.

- 10 Выберите свой запрос в окне списка, затем нажмите OK.



11. Отобразится Design Tab с диалоговым окном Insert Fields. В Database Tab диалогового окна Insert Fields появится Ваш запрос Access и все связанные с этим запросом поля.

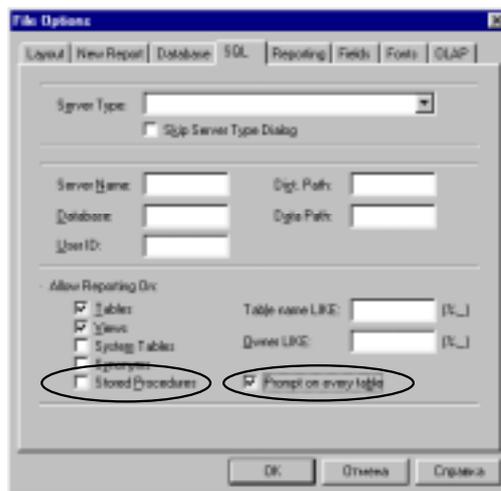
ВНИМАНИЕ: В Crystal Report Designer нельзя использовать запросы действия или обновления Access. Вы можете использовать только запросы выбора и комбинационных таблиц.

Как открывать запросы параметров Access

Запросы параметров Access могут быть открыты только в том случае, если база данных Access открыта через ODBC. Перед выполнением данной процедуры удостоверьтесь, что для Вашей базы данных Access установлен источник данных ODBC. Смотрите: *Как устанавливать источник данных ODBC.*

ВНИМАНИЕ: Если Вы создаете запрос параметра в Access, Вы должны обеспечить приглашение для запроса и задать тип данных для параметра. Сначала, когда Ваш запрос открыт в Design View в Microsoft Access, введите приглашение в ячейке Criteria для поля, которое будет выполнять функцию параметра. Затем выберите команду Parameters из меню Query в Access и задайте тип данных для только что созданного Вами параметра. Удостоверьтесь, что приглашение отобразилось в том же виде, что и в ячейке Criteria. Полные инструкции приведены в документации к Access. Если Вы установите запрос параметра некорректно, Crystal Report Designer не сможет его использовать.

1. В Crystal Report Designer выберите команду **Options** из меню File. Отобразится диалоговое окно File Options. Чтобы его инициировать, щелкните по SQL Tab.

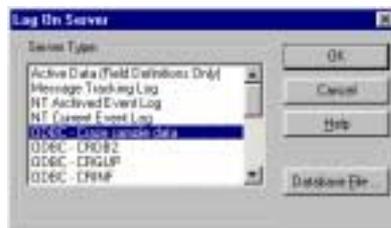


- 2 Удостоверьтесь, что поставлены флажки Stored Procedures или Prompt on every table

- Опция *Stored Procedures* автоматически отображает все доступные сохраненные процедуры, если Вы логинуетесь к источнику данных ODBC. Crystal Report Designer воспринимает запросы параметров ODBC почти так же, как он воспринимает сохраненные процедуры SQL. Поэтому для использования запроса параметра должна быть подключена опция *Stored Procedures*.
- Опция *Prompt on every table* отображает диалоговое окно Allow Reporting On каждый раз, когда Вы присоединяетесь к источнику данных ODBC. Это диалоговое окно поддерживает те же опции, что и раздел *Allow Reporting On* диалогового окна

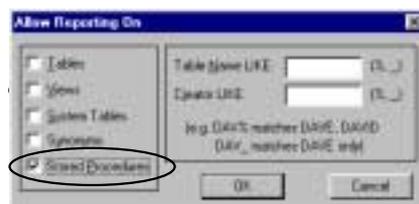
File Options, но оно поддерживает эти опции для каждого источника данных ODBC, к которому Вы логируетесь.

3. Кроме того, если хотите, Вы можете задать опции *Table Name LIKE* и *Owner LIKE*.
 - *Table Name LIKE* основана на выражении SQL LIKE. Эта опция позволяет Вам задавать виды имен таблиц, которые должны отображаться в диалоговом окне Choose SQL Table. В качестве специальных символов используйте с данной функцией символ подчеркивания (_) или знак процента (%). Символ подчеркивания задает один символ, а знак процента – строку символов. Например, если Вы зададите DAV_ , то отобразится только DAV, а если зададите DAV%, отобразятся DAVE и DAVID. *Table Name LIKE C%* будет отображать только те таблицы, имена которых начинаются с буквы C.
 - *Owner LIKE* также основана на выражении SQL LIKE. Опция *Owner LIKE* позволяет Вам выбрать Owner (Владельца) (или Creator (Создателя) или Alias (Псевдоним)) таблицы, а не само имя таблицы. Например, *Owner LIKE C%* отобразит только те таблицы, имя владельца которых начинается в буквы C.
4. Нажмите *OK* для выхода из диалогового окна File Options.
5. Создайте новый отчет. При выборе Вами SQL/ODBC в качестве источника данных на экране отобразится диалоговое окно Log On Server.



6 Выберите источник данных ODBC для своей базы данных ACCESS, затем нажмите OK.

7. Если Вы не задали определенный файл базы данных Access в источнике данных ODBC Access, то на экране отобразится диалоговое окно Select Database. Найдите и выберите базу данных, содержащую запрос параметра Access, который Вы хотите использовать, затем нажмите *OK*.
8. Если в диалоговом окне File Options Вы выбрали флажок *Prompt on every table*, то отобразится диалоговое окно Allow Reporting On. В остальных случаях перейдите к Шагу 11.



9 Удостоверьтесь, что флажок Stored Procedures установлен, затем нажмите OK.

10. Отобразится диалоговое окно Choose SQL Table. В окне списка SQL Tables отобразится запрос параметра Access в следующем виде:

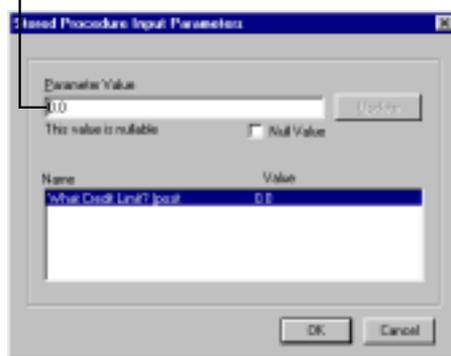
Proc (QueryName)

11 Выберите свой запрос в окне списка, затем нажмите ОК.



Отобразится диалоговое окно Stored Procedures Input Parameters.

12 Выделите параметр в этом окне списка.



13 Присвойте значение, введя его в текстовое окно Parameter Value, и щелкните по кнопке Update.

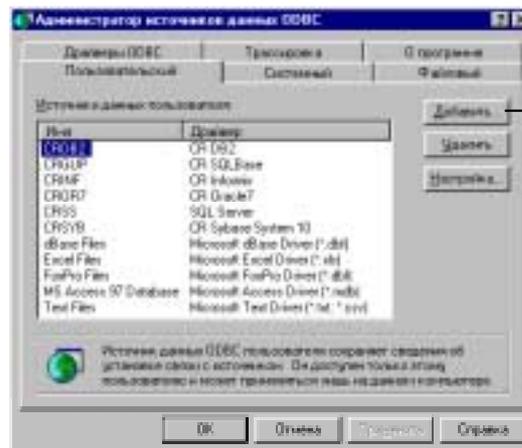
14. Значение, отображенное в окне списка *Name*, обновится.
15. Повторите Шаги 12 и 13 для каждого параметра в Вашем запросе параметров Access.
16. После завершения нажмите на *ОК*. Вы можете изменять значения параметров в любое время с помощью команды **Stored Procedure Parameters** в меню Database.
17. Создайте свой отчет, используя поля в запросе параметров. В вашем отчете будут использованы только те записи, которые удовлетворяют значениям параметров, заданным Вами в диалоговом окне Stored Procedure Input Parameters.

ВНИМАНИЕ: В *Crystal Report Designer* нельзя использовать запросы действия или обновления Access. Вы можете использовать только запросы выбора и комбинационных таблиц.

Как устанавливать источник данных ODBC

Для установки источника данных ODBC у Вас должен быть установлен драйвер ODBC для того типа данных, который Вы собираетесь использовать. Многие приложения DBMS автоматически устанавливают и устанавливают драйверы ODBC. Если Вы не уверены в том, что для Ваших данных были установлены драйверы ODBC, обратитесь за этой информацией к документации для Вашего приложения DBMS.

1. Из Program Manager щелкните дважды мышкой по иконке ODBC Administrator в соответствующей группе программ. На экране отобразится диалоговое окно ODBC Data Source Administrator.



2 Чтобы добавить новый источник данных ODBC щелкните по кнопке Добавить (Add).

Отобразится диалоговое окно Add Data Source.



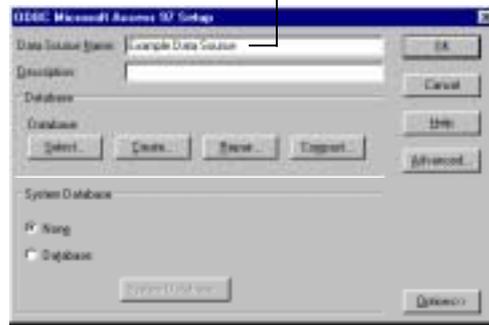
- 3 Выберите из данного списка драйвер ODBC, соответствующий Вашему типу данных

Если в списке не появится драйвер для Вашего типа данных, значит драйвер ODBC был установлен некорректно. Обратитесь к документации для Вашего приложения DBMS.

4. Когда Вы нажмете на *OK*, отобразится диалоговое окно ODBC Data Source Setup, специфичное для выбранного Вами драйвера ODBC.

ВНИМАНИЕ: Если вместо диалогового окна *Setup* отобразится сообщение об ошибке, это означает, что Вы установили в систему драйверы ODBC, не соответствующие выбранному Вами типу данных.

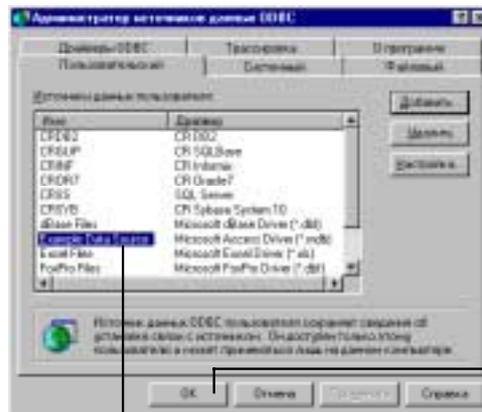
5 Введите в указанном месте имя своего нового источника данных ODBC



Это имя, которое Вы выбираете при логировании к источнику данных из Crystal Report Designer.

ВНИМАНИЕ: Внешний вид отображенного диалогового окна может отличаться от приведенного в настоящем документе, т.к. он зависит от выбранного Вами типа данных. Показанное здесь диалоговое окно относится к драйверу ODBC Access 7.0. Для получения полной информации об использовании отображаемого для Вашего типа данных диалогового окна щелкните мышкой по кнопке Help.

6. После завершения установки своего источника данных нажмите на **OK** в диалоговом окне ODBC Data Source Setup. Новый источник данных ODBC появится в диалоговом окне Data Sources.

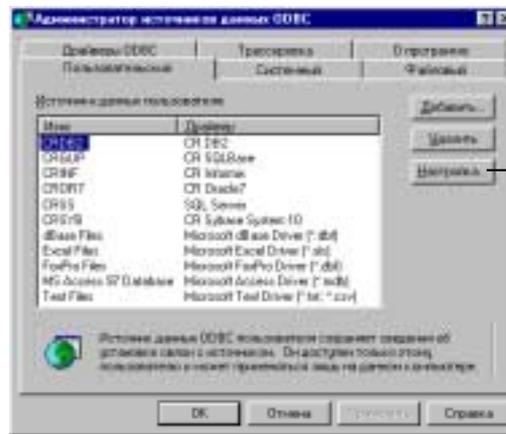


7 После завершения щелкните по кнопке **OK**

Теперь в данном списке появился новый источник данных

Как проверять установки для источника данных ODBC

1. Из соответствующей группы программ или папки в Windows 95 запустите приложение Администратор ODBC. На экране отобразится диалоговое окно Data Sources.
2. Выделите из появившегося списка соответствующий источник данных.



3 Щелкните по кнопке Настройка (Configure).

На экране отобразится диалоговое окно ODBC Data Source Setup.



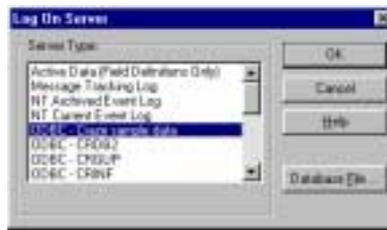
4. Диалоговое окно ODBC Data Source Setup специфично для каждого выбранного источника данных и содержит элементы управления и информацию для установки источника данных. Проверьте установки в этом диалоговом окне на предмет их соответствия Вашей системе и базе данных.

ВНИМАНИЕ: Отображенное у Вас диалоговое окно может отличаться от приведенного в данном примере, т.к. оно зависит от используемого типа данных. Показанное диалоговое окно относится к драйверу ODBC Access 7.0. Для получения дополнительной информации об использовании диалогового окна для Вашего типа данных щелкните мышкой по кнопке Help.

5. Сделайте все необходимые изменения и нажмите OK.
6. Закройте диалоговое окно Администратора ODBC Data Sources.

Как логироваться к источнику данных ODBC

1. Выберите команду **Log On Server** из меню Database. На экране отобразится диалоговое окно Log On Server.



- 2 Выберите источник данных ODBC, соответствующий файлу базы данных, который Вы собираетесь открыть, затем нажмите OK.

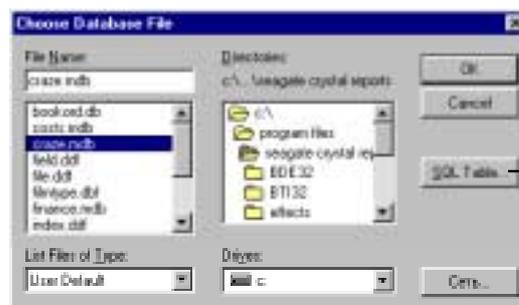
3. Если источник данных требует пароля или имени пользователя, или какую-либо другую информацию логирования, то отобразится диалоговое окно логирования. Введите информацию, которую обычно используете для доступа к этой базе данных, затем нажмите *OK*.

- Если при задании источника данных ODBC Вы не задали базу данных, то на экране отобразится диалоговое окно Select Database. Выделите нужный файл базы данных с помощью элементов управления *Drives*, *Directories* и *Database Name*, затем нажмите *OK*.

4. На экране отобразится диалоговое окно Choose SQL Table. Выберите таблицу базы данных из списка *SQL Tables* и нажмите **OK**, чтобы добавить эту таблицу в Ваш отчет.

Как добавлять таблицу базы данных ODBC в отчет

1. Выберите команду **Add Database To Report** из меню Database. Отобразится диалоговое окно Choose Database File.



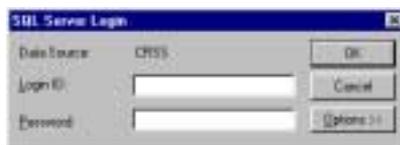
- 2 Щелкните по кнопке SQL Table.

На экране отобразится диалоговое окно Log On Server.



- 3 Выберите источник данных ODBC, соответствующий файлу базы данных, который Вы собираетесь открыть, затем нажмите OK.

4. Если источник данных требует пароля или имени пользователя, или какую-либо другую информацию логирования, то отобразится диалоговое окно SQL Server Login.



5 Введите необходимую информацию, затем, когда завершите логирование, нажмите OK.

6. Если выбранный Вами источник данных включает спецификацию файла базы данных, или если Вы задали в Шаге 3 таблицу базы данных, перейдите к Шагу 8.
7. На экране отобразится диалоговое окно Select Database. Выделите нужный файл базы данных с помощью элементов управления *Drives, Directories* и *Database Name*, затем нажмите **OK**.
8. На экране отобразится диалоговое окно Choose SQL Table.
 - Выберите таблицу базы данных из списка *SQL Tables* и нажмите **OK**, чтобы добавить ее в Ваш отчет, или
 - Нажмите на кнопку *Log On Server*, чтобы логироваться к другому источнику данных ODBC.

Как присоединиться к Серверу MS SQL через ODBC

ВНИМАНИЕ: Настоящий раздел приведен в качестве примера того, как логироваться к источнику данных Сервера Microsoft SQL из Crystal Report Designer. Ваше приложение сервера SQL или другой защищенный паролем источник данных может потребовать выполнения других шагов. Приведенный ниже пример относится только к одному из типов баз данных SQL.

1. Проверьте установки источника данных Сервера Microsoft SQL с помощью Администратора ODBC. Смотрите *Как проверять установки источника данных ODBC*.
2. Из Crystal Report Designer выберите команду **Log On Server** из меню Database. На экране отобразится диалоговое окно Log On Server.



3 Выберите источник данных ODBC - CRSS, затем нажмите OK.

Источник данных ODBC-CRSS создается автоматически во время проведения процедуры установки Crystal Report Designer. Этот источник данных позволяет Вам открывать базы данных Сервера MS SQL.

На экране отобразится диалоговое окно SQL Server Login.



4 Введите ID логирования и пароль для своего сервера MS SQL.

5 Затем щелкните по кнопке Options.

На экране отобразится раздел Options диалогового окна SQL Server Login.



6 Проверьте установки для своего сервера SQL.

7. Проверьте имя базы данных Сервера SQL, которую Вы собираетесь открыть, язык базы данных, который хотите использовать, имя используемого Вами приложения (Crystal Report Designer), а также имя Вашей компьютерной рабочей станции. Внесите все необходимые изменения.
8. Нажмите **ОК**. На экране отобразится диалоговое окно Choose SQL Table. Выберите таблицу базы данных Сервера SQL из списка *SQL Tables* и нажмите **ОК**, чтобы добавить эту таблицу в Ваш отчет.

Как отсоединяться от источника данных ODBC

1. Выберите команду **Log Off Server** из меню Database. На экране отобразится диалоговое окно Log Off Server.
- 2 В появившемся списке выделите источник данных ODBC, от которого Вы хотите отсоединиться.

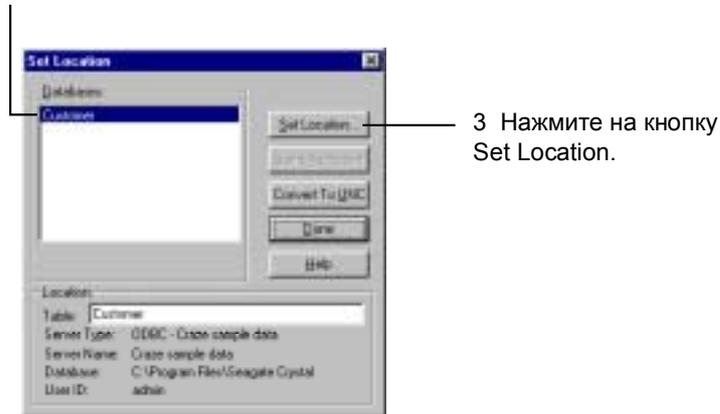


3 Затем нажмите на кнопку Log Off.

4. Источник данных ODBC исчезнет из списка *Logged On Servers*.
5. После завершения нажмите на кнопку **Done**.

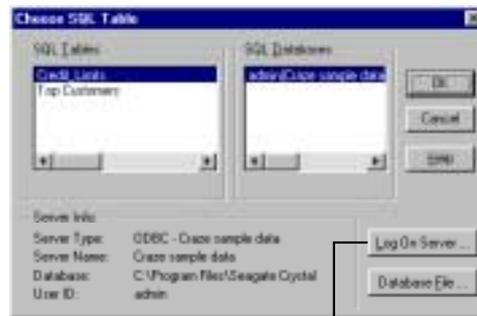
Как изменить источник данных ODBC, доступный для отчета

1. Выберите команду **Set Location** из меню Database. На экране отобразится диалоговое окно Set Location со списком псевдонимов таблиц отображенного отчета. Под списком таблиц будет находиться информация о расположении и логировании выделенных Вами таблиц. Выберите первую таблицу в списке и нажмите на кнопку **Set Location**.
2. Выделите первую таблицу в списке.



4. Если Вы еще этого не сделали, присоединитесь к оригинальному источнику данных ODBC. Сначала Вы должны присоединиться к старому источнику данных, и только после этого Вы сможете изменить отчет, чтобы он начал использовать новый источник данных. Так как Вы должны быть логированы к старому источнику данных, он должен быть доступен на локальной машине.
 - Если машина уже не присоединена к старому серверу, Вы можете установить источник данных ODBC с именем старого источника данных и сделать так, чтобы он указывал на новый сервер базы данных. Присоединитесь к этому источнику данных как к "старому" источнику данных.
 - Если для присоединения к новому серверу Вы используете то же имя источника данных, Вы должны сконфигурировать этот источник под ODBC Admin или ODBC Control Panel и сделать так, чтобы он указывал на новый сервер.

После логирования к старому источнику данных на экране отобразится диалоговое окно Choose SQL Table со списком таблиц в окне списка *SQL Tables* и старым источником данных в окне списка *SQL Database*.

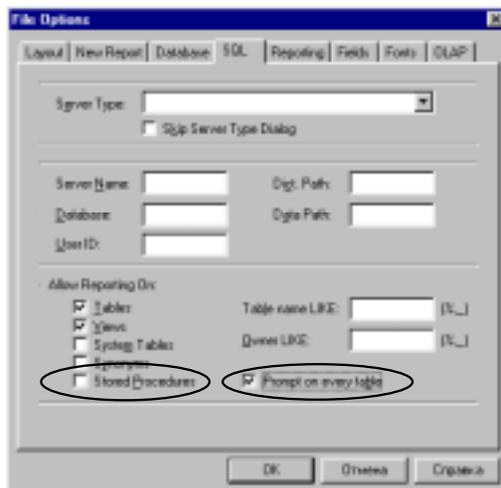


5 Нажмите на кнопку Log On Server для логирования к новому источнику данных.

- Теперь Вы должны быть присоединены к обоим источникам данных: старому и новому.
 - Новый источник данных должен появиться в окне списка *SQL Database* диалогового окна Choose SQL Table.
6. Щелкните мышкой по новому источнику данных, чтобы отобразились его таблицы, затем из появившегося списка выберите нужную таблицу. Выбираемая Вами таблица должна соответствовать первой таблице, выбранной Вами в диалоговом окне Set Location.
 7. Нажмите **ОК**. На экране появится приглашение к изменению положения всех таблиц отчета на то местоположение, которое Вы задали для первой таблицы.
 8. В появившемся окне щелкните по **Yes**, чтобы установить местоположение всех таблиц отчета в одной и той же базе данных.
 9. Закройте диалоговое окно Set Location, затем выберите команду **Verify Database** из меню Database, чтобы обновить определения таблиц в отчете.
 10. Отсоединитесь от старого источника данных.
 11. Сохраните отчет.

Как выбирать сохраненную процедуру из базы данных SQL и изменять параметры сохраненных процедур

1. Выберите команду **Options** из меню Files. На экране отобразится диалоговое окно File Options. Чтобы его активизировать, нажмите на **SQL Tab**.

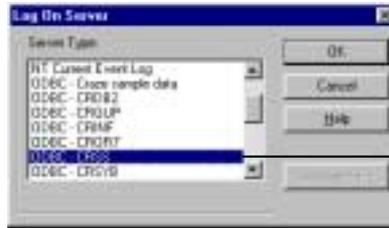


2. Удостоверьтесь, что поставлены флажки **Stored Procedures** или **Prompt on every table**

- Опция *Stored Procedures* автоматически отображает все доступные сохраненные процедуры, когда Вы логируетесь к базе данных SQL.
 - Опция *Prompt on every table* отображает диалоговое окно *Allow Reporting On* каждый раз, когда Вы присоединяетесь к источнику данных ODBC. Это диалоговое окно поддерживает те же опции, что и раздел *Allow Reporting On* диалогового окна *File Options*, но предоставляет эти опции для каждого источника данных ODBC, к которому Вы логируетесь.
3. Кроме того, если хотите, Вы можете задать опции *Table Name LIKE* и *Owner LIKE*.
 - *Table Name LIKE* основана на выражении SQL LIKE. Эта опция позволяет Вам задавать виды имен таблиц, которые должны отображаться в диалоговом окне *Choose SQL Table*. В качестве специальных символов используйте с данной функцией символ подчеркивания () или знак процента (%). Символ подчеркивания задает один символ, а знак процента – строку символов. Например, если Вы зададите DAV_, то отобразится только DAV, а если зададите DAV%, отобразятся DAVE и DAVID. *Table Name LIKE C%* будет отображать только те таблицы, имена которых начинаются с буквы C.
 - *Owner LIKE* также основана на выражении SQL LIKE. Опция *Owner LIKE* позволяет Вам выбрать Owner (Владельца) (или Creator (Создателя) или Alias (Псевдоним)) таблицы, а не само имя таблицы. Например, *Owner LIKE C%* отобразит только те таблицы, имя владельца которых начинается в буквы C.

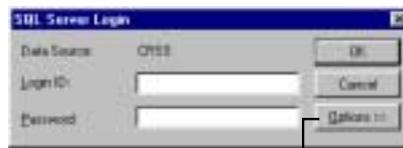
ВНИМАНИЕ: Для получения более полной информации о выражении *SQL LIKE* обратитесь к документации SQL.

4. Нажмите **ОК** для выхода из диалогового окна File Options.
5. Создайте новый отчет на основе базы данных SQL, содержащей сохраненную процедуру, которую Вы собираетесь использовать. Если в качестве источника данных Вы выберете SQL/ODBC, на экране отобразится диалоговое окно Log On Server.



6 Выделите из данного списка источник данных ODBC для своего сервера SQL, затем нажмите ОК.

На экране отобразится диалоговое окно SQL Server Login.



7 Введите необходимую информацию, затем, после завершения логирования, нажмите ОК.

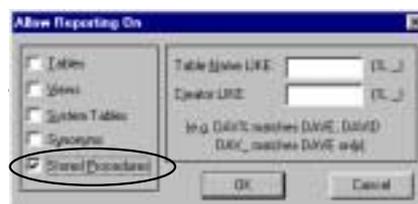
8 Щелкните по кнопке Options.

На экране отобразится раздел Options диалогового окна SQL Server Login.



9 Проверьте установки для своего сервера SQL.

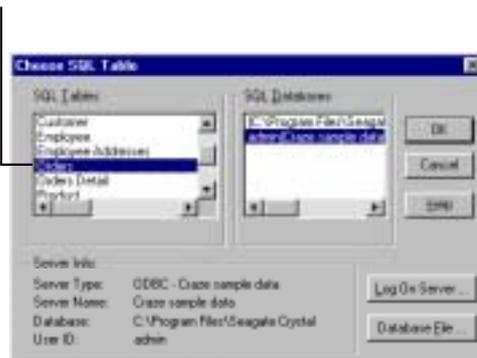
10. Нажмите **ОК**. Если в диалоговом окне File Options у Вас установлен флажок *Prompt on every table*, то на экране отобразится диалоговое окно Allow Reporting On. Если же нет, перейдите к Шагу 12.



11 Удостоверьтесь, что флажок Stored Procedures установлен, затем нажмите ОК.

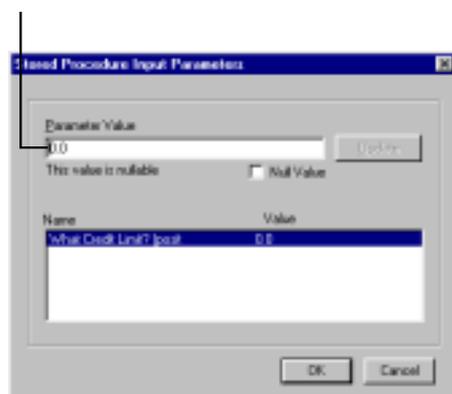
На экране отобразится диалоговое окно Choose SQL Table.

12 Выделите из данного списка сохраненную процедуру SQL, затем нажмите OK.



13. Сохраненная процедура будет добавлена в Ваш отчет, и на экране отобразится диалоговое окно Stored Procedure Input Parameters.

14 Выделите параметр в этом окне списка.



15 Присвойте значение, введя его в текстовое окно Parameter Value, и щелкните по кнопке Update.

16. Значение, отображенное в окне списка *Name*, обновится.

17. Повторяйте шаги 14 и 15 для каждого параметра, отображенного в окне списка *Name/Value*.

18. После завершения нажмите **OK**. Вы можете изменять значения параметров в любое время с помощью команды **Stored Procedure Parameters** из меню Database.

ИСТОЧНИКИ ДАННЫХ

Введение

Crystal Report Designer может иметь доступ к данным, сохраненным почти в любом обычном формате данных, а также во многих нестандартных форматах. Кроме того, Crystal Report Designer может использовать все преимущества, предоставляемые файлами запросов (.QRY) и словарями (.DC5). Смотрите: *Queries* и *Dictionaries*.

В настоящей главе описаны различные типы данных, доступ к которым имеет Crystal Report Designer, а также дано объяснение уровней доступа к данным, используемым при присоединении к данным. Если Вы не знаете, какую Систему управления базами данных (DBMS) использует Ваша компания, свяжитесь с менеджером IS или сетевым администратором.

Почему необходимо прочесть данную главу?

Основное назначение Crystal Report Designer заключается в доступе к данным, сохраненным в базе данных, и создании на основе этих данных отчетов. Такое применение – одно из старейших в использовании компьютеров и до настоящего времени остается наиболее частым и необходимым. Crystal Report Designer разрабатывался с целью упрощения решения данной задачи, снижения временных и трудовых затрат.

В основе создания любого отчета лежит идея доступа к данным. Зная, как Crystal Report Designer осуществляет доступ к данным, Вы сможете лучше понять процесс создания отчетов, а также лучше узнать типы данных, с которыми может работать программа.

Кроме того, понимание процесса доступа к данным поможет Вам решить проблемы, с которыми Вы можете столкнуться при попытке открыть файл данных. Это особенно полезно для менеджеров IS и тех, кто обеспечивает поддержку доступа к данным для групп пользователей.

Большая часть информации, представленной в настоящей главе, предназначена для опытных пользователей Crystal Report Designer и менеджеров IS; она охватывает технические аспекты Систем управления базами данных (DBMS) и методы сохранения данных. Предполагается знание компьютера, операционной системы Windows, Windows 95 или Windows NT, а также хотя бы одного из приложений DBMS.

Вниманию разработчиков

В настоящей главе описаны принципы доступа к данным. Однако эти же концепции применимы к любому приложению, осуществляющему доступ к данным посредством Crystal Report Designer или любых инструментов разработки Crystal Report Engine, описанных в Техническом справочнике. Поэтому использование Crystal Report Designer, описанное в данной главе, относится как к приложению, так и к Report Engine, если не оговорено обратное.

ВНИМАНИЕ: Большинство имен файлов, упомянутых в данной главе, относится к 16-битовой версии программы, если не оговорено противное. Для 32-битовой версии используются аналогичные имена файлов, но в них должно присутствовать указание на их принадлежность 32-битовой программе. Например, PDSODBC.DLL является 16-битовым трансляционным файлом ODBC, а P2SODBC.DLL – 32-битовым трансляционным файлом ODBC.

Четыре типа данных

Типы данных, доступные для Crystal Report Designer, распадаются на четыре основных категории:

1. Файлы базы данных с прямым доступом.
2. Источники данных ODBC.
3. Файлы Crystal Query Designer.
4. Файлы Crystal Dictionary.

Каждый тип данных становится доступным при использовании определенного набора DLL и других соответствующих файлов доступа к данным. Поняв процесс, используемый программой для доступа к данным, Вы сможете лучше понять процесс создания отчета и элементы, используемые для превращения имеющихся в Вашем распоряжении данных в прекрасные отчеты. Смотрите: *Queries* и *Dictionaries*.

ВНИМАНИЕ: При осуществлении доступа к любому типу данных Crystal Report Designer получает информацию об именах файлов, типах полей и длине полей от драйверов баз данных. Эта информация предоставляется механизмом базы данных или драйвером базы данных ODBC.

Файлы базы данных с прямым доступом

Crystal Report Designer может осуществлять прямой доступ к большинству обычных форматов баз данных PC. Другими словами, программа имеет встроенные возможности для прямого открытия файлов и таблиц базы данных, созданных в dBASE, FoxPro, Clipper, Vtrieve, Paradox и Microsoft Access и многих других. Инсталлировав Crystal Report Designer, Вы получаете эту функциональную возможность. После инсталляции Crystal Report Designer Вы сразу же можете начать создавать отчеты на основе этих баз данных путем простого выбора соответствующего файла.

Преимущества

Прямой доступ к базе данных является самым быстрым способом прочтения данных. Crystal Report Designer должен будет связаться только с одним уровнем доступа к данным, обеспечивающим связь с данными. Результаты будут получены быстро практически в любом системном окружении.

Кроме того, доступ к данным прост. Файлы базы данных с прямым доступом являются источниками данных "point-and-click" ("указал и щелкнул"). Вам нужно только выбрать требуемые файлы базы данных, и Crystal Report Designer прочтет все сохраненные данные.

Недостатки

Если Вы получаете прямой доступ к базе данных из Crystal Report Designer, то отчет сможет использовать только данный тип базы данных. Вы не сможете переключиться на другой тип базы данных или таблицы без создания нового отчета.

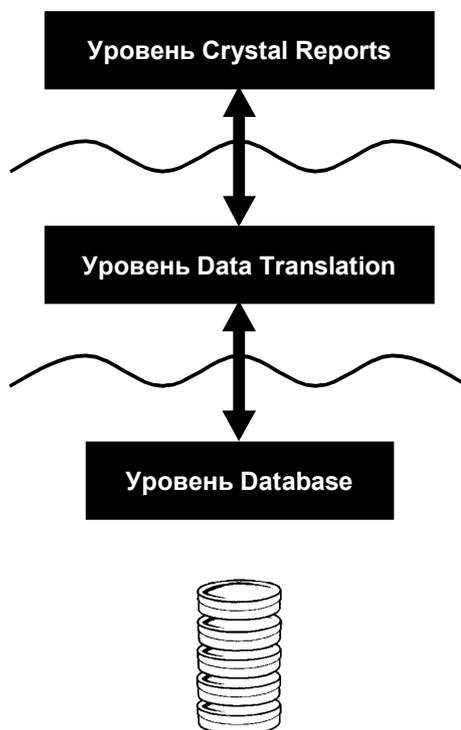
Например, если Вы создаете отчет на основе данных Btrieve, Вы не сможете изменить таблицы, к которым отчет имеет доступ, на данные Paradox. Crystal Report Designer связывается с данными Btrieve, используя специальный синтаксис Btrieve, а данный синтаксис не совместим с данными Paradox.

С другой стороны, если Вы осуществляете доступ к данным через ODBC, используемый синтаксис всегда представляет собой язык SQL, независимо от фактического типа базы данных. Смотрите *Источники данных ODBC*.

Три уровня

Прямой доступ к файлам базы данных из Crystal Report Designer требует трех уровней:

1. Crystal Report Designer.
2. Data Translation.
3. Database.



Crystal Report Designer использует уровень трансляции данных для обращения к базе данных и доступа к ее данным.

Crystal Report Designer

Crystal Report Designer действует как интерфейс, посредством которого Вы форматируете, организуете, выбираете и сортируете данные, хранящиеся в файлах базы данных. Он получает данные путем установления связи с одним или несколькими файлами в трансляционном уровне данных, который может фактически читать базу данных. Так как Crystal Report Designer может работать со многими различными формами данных, для непосредственной работы с данными он должен опираться на другие файлы. Тогда Crystal Report Designer может использовать собственный метод связи для общения с трансляционными файлами.

Data Translation

Данные транслируются через набор DLL, специфических для Crystal Report Designer. Для понимания того, как организованы данные определенного типа, и корректного их представления во время печати, просмотра или экспортирования отчета программа использует DLL, относящиеся к этому типу данных.

ВНИМАНИЕ: *Crystal Report Designer поставляется в комплекте со всеми файлами транслирования данных для каждого поддерживаемого им типа базы данных с прямым доступом. Для получения более полной информации обо всех необходимых файлах обратитесь к оперативной подсказке Runtime File Requirements (RUNTIME.HLP).*

Database

Файл базы данных состоит из одной или нескольких таблиц. Различные приложения DBMS по-разному хранят информацию баз данных. Например, dBASE хранит каждую таблицу базы данных как отдельный файл. Access же может хранить несколько таблиц, вместе с запросами, макрос и другими элементами базы данных, в одном файле.

Когда Crystal Report Designer осуществляет прямой доступ к файлу базы данных, он автоматически ищет информацию обо всех таблицах и полях в этом файле. Возможно, Вы не будете использовать все таблицы и поля, но программа сделает их все доступными для Вас. Другими словами, если открыт файл dBASE, доступной будет только одна таблица в этом файле. Однако, если открыт файл Access, то доступны будут все таблицы этого файла, если даже Вы никогда не будете их использовать.

ВНИМАНИЕ: *Crystal Report Designer также будет открывать запросы в базе данных Access, используя для этого механизм DAO или ODBC, и позволит Вам использовать для отчета поля запросов так же, как Вы используете поля таблиц. Смотрите: DAO и Access.*

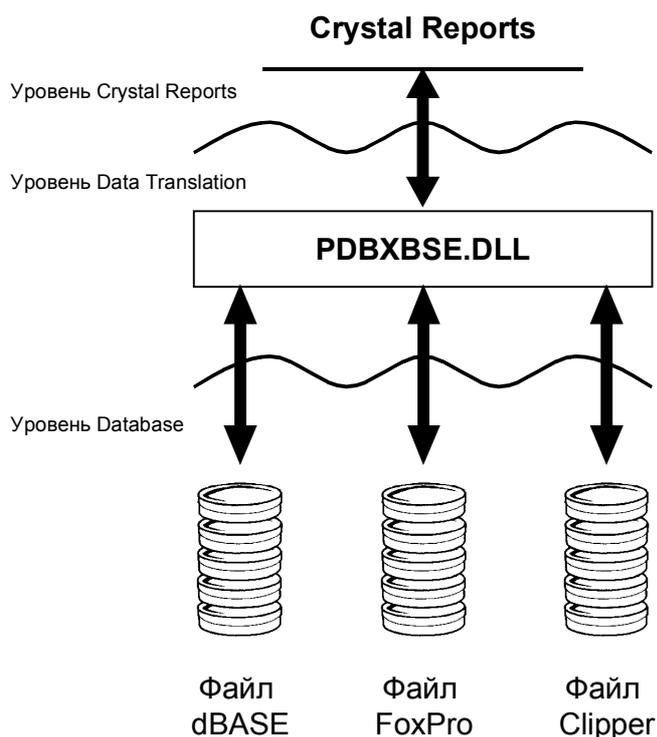
Обычные форматы баз данных

Хотя для получения данных из всех форматов файлов баз данных с прямым доступом Crystal Report Designer использует ту же систему с тремя уровнями связей, каждый формат требует отдельного набора Библиотек Динамической Связи (DLL). Однако некоторые форматы расширяют эту базовую структуру с тремя уровнями связей. В последующих разделах описана система, которую Crystal Report Designer использует для доступа к данным из некоторых наиболее популярных форматов баз данных.

dBASE, FoxPro, Clipper

Формат баз данных dBASE остается одной из наиболее популярных систем управления базами данных (DBMS), используемых в бизнесе. В связи с этим, Crystal Report Designer создавался с учетом обеспечения возможности непосредственно и просто открывать данные dBASE с помощью механизма xBase (в PDBXBSE.DLL). FoxPro и Clipper являются форматами баз данных, совместимыми с dBASE, поэтому для доступа к любому из трех указанных приложений DBMS Crystal Report Designer использует одну и ту же DLL.

ВНИМАНИЕ: Трансляционный уровень PDBXBSE.DLL поддерживает файлы FoxPro до версии 2.5. Смотрите Visual FoxPro.



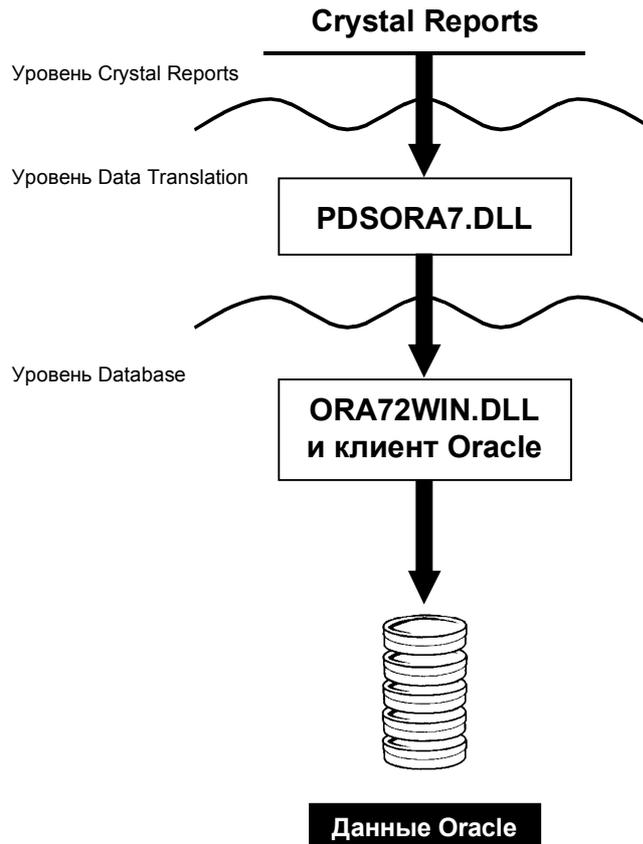
Файл PDBXBSE.DLL осуществляет всю трансляцию между Crystal Report Designer и файлами dBASE, FoxPro или Clipper. Каждый файл базы данных содержит только одну таблицу базы данных, но, при этом, отчет может иметь доступ к неограниченному количеству файлов.

ВНИМАНИЕ: Доступ к данным dBASE может также осуществляться посредством Механизма базы данных Borland (BDE) с использованием PDBBDE.DLL. Чтобы узнать, как BDE связывается с данными базы данных, смотрите Paradox. Однако BDE не поддерживает данные FoxPro или Clipper.

Oracle 7

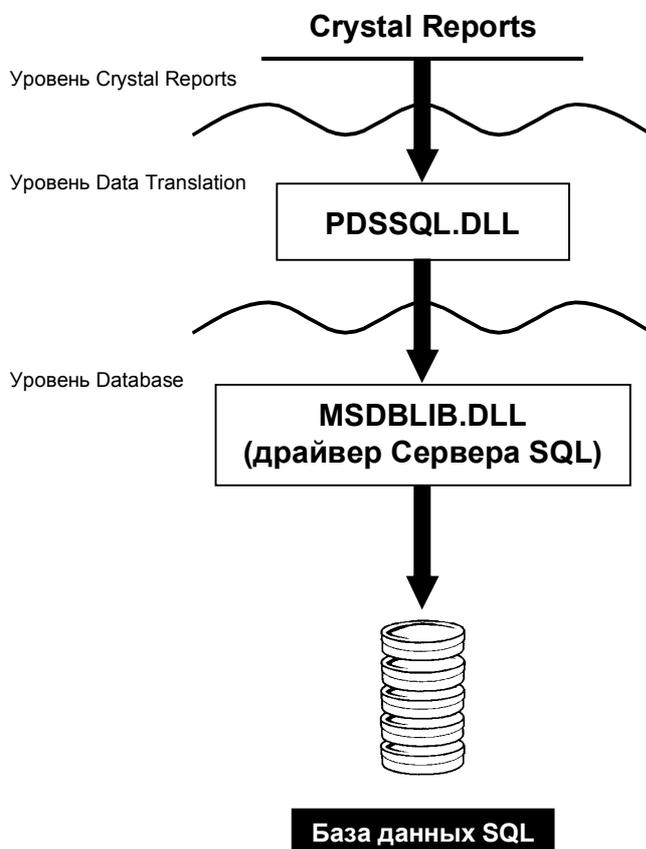
Настоящая версия Crystal Report Designer поддерживает прямой доступ к базам данных SQL Oracle 7. Такой доступ обеспечивается трансляционным файлом PDSORA7.DLL (инсталлируемым с программой). Данный файл может непосредственно связываться с драйвером базы данных Oracle 7 ORA72WIN.DLL, который работает прямо с базами данных Oracle и клиентами, проводя поиск необходимой для отчета информации.

ВНИМАНИЕ: Клиентское программное обеспечение Oracle должно быть инсталлировано в Вашу систему, файл ORA72WIN.DLL должен находиться в операторе PATH Вашего файла AUTOEXEC.BAT.



Сервер SQL Microsoft 6.x

Crystal Report Designer может непосредственно считывать базы данных, созданные Сервером SQL Microsoft версии 6.0 и 6.5. Файл PDSSQL.DLL, установленный с программой, транслирует запросы на сервер SQL и возвращает с него данные. Он осуществляет прямую связь с MSDBLIB.DLL драйвера Сервера SQL Microsoft.



ВНИМАНИЕ: Сетевые администраторы и менеджеры IS, работающие с Сервером Системного Управления Microsoft для BackOffice, могут составлять отчеты из данных Сервера Системного Управления. Сервер Системного Управления использует Сервер SQL для хранения системных данных, поэтому файлы, используемые для доступа к этим данным, аналогичны приведенным здесь файлам для Сервера SQL.

Источники данных ODBC

Открытая связь с базами данных (ODBC) представляет собой стандарт, разработанный корпорацией Microsoft, с помощью которого одно приложение может получать доступ к самым различным типам данных. Приложению необходимо только установить связь с одним набором файлов, ODBC, и оно сразу же получит возможность работать с любым источником данных, доступным по ODBC.

Существуют сотни Систем управления базами данных (DBMS) для персональных компьютеров и тысячи приложений, осуществляющих доступ к данным DBMS. Обычно компания, разрабатывающая приложение, осуществляющее доступ к данным, такое как Crystal Report Designer, должна разрабатывать средства для осуществления связи между приложением и каждым типом данных, использование которого может потребоваться заказчику. Crystal Report Designer делает это посредством прямого доступа к базам данных.

С другой стороны, если DBMS просто обеспечивает средства, с помощью которых ODBC может получать доступ к ее данным, то данные DBMS становятся источником данных ODBC. Любое приложение, такое как Crystal Report Designer, может устанавливать связь с ODBC и мгновенно получать доступ к любому источнику данных ODBC. Благодаря наличию драйверов ODBC для большинства продуктов DBMS, Crystal Report Designer имеет почти неограниченные возможности по доступу к различным типам данных.

Преимущества

Возможно, самым большим преимуществом доступа к данным через ODBC является возможность доступа к широкому диапазону данных с помощью одного интерфейса. Так как теперь наиболее популярные Системы управления данными предлагают драйверы ODBC, которых с каждым днем появляется все больше и больше, Crystal Report Designer может использовать любой имеющийся у Вас тип данных.

Благодаря большой гибкости ODBC, Вы можете использовать один и тот же файл отчета с различными источниками данных ODBC. Например, Вы создали отчет, используя источник данных Oracle, затем, когда Ваша компания переключилась на Сервер SQL Microsoft, Вы можете просто изменить источник данных, используемый Вашим отчетом. Единственное требование заключается в том, чтобы новый источник данных имел ту же структуру (таблицы и поля), что и оригинальный источник данных (хотя имена таблиц могут и отличаться). Смотрите *Как изменять источник данных ODBC, к которому отчет осуществляет доступ*.

Опытные программисты SQL (Язык структурированных запросов) также получают преимущества от стандарта ODBC. Так как Crystal Report Designer использует SQL для связи с ODBC, программисты SQL и Администраторы баз данных могут просматривать и редактировать оператор SQL, посылаемый на ODBC, управляя таким образом тем, как осуществляется поиск данных в источнике данных.

И, наконец, используя технологию передачи SQL для отправки оператора на ODBC и поиска начального набора данных, Crystal Report Designer перекладывает большую часть работы по поиску и сортировке данных на систему сервера, освобождая локальную память и ресурсы для решения более важных задач. Кроме того, в Crystal Report Designer

возвращаются только те данные, которые заданы оператором SQL, что снижает сетевой трафик и использование сетевых ресурсов. Более эффективно работая с оригинальными данными, Crystal Report Designer экономит Ваше время и силы и позволяет использовать их для процесса разработки и другой более важной работы.

Недостатки

Так как в процесс передачи данных из базы данных в приложение по ODBC вовлечено много уровней, источнику данных ODBC часто может потребоваться больше времени, чтобы вернуть данные. Сначала Crystal Report Designer должен запросить некоторые данные. Трансляционный уровень ODBC должен транслировать запрос в формат, понятный для ODBC (оператор SQL). ODBC должен определить, где размещаются запрошенные данные, и передать запрос в источник данных ODBC; смотрите *Трансляционный уровень DBMS (источник данных ODBC)*. Источник данных должен проанализировать запрос и снова транслировать его в формат, понятный для DBMS. Этот сложный процесс не только занимает довольно много времени, но и при его выполнении на любом из уровней может произойти ошибка.

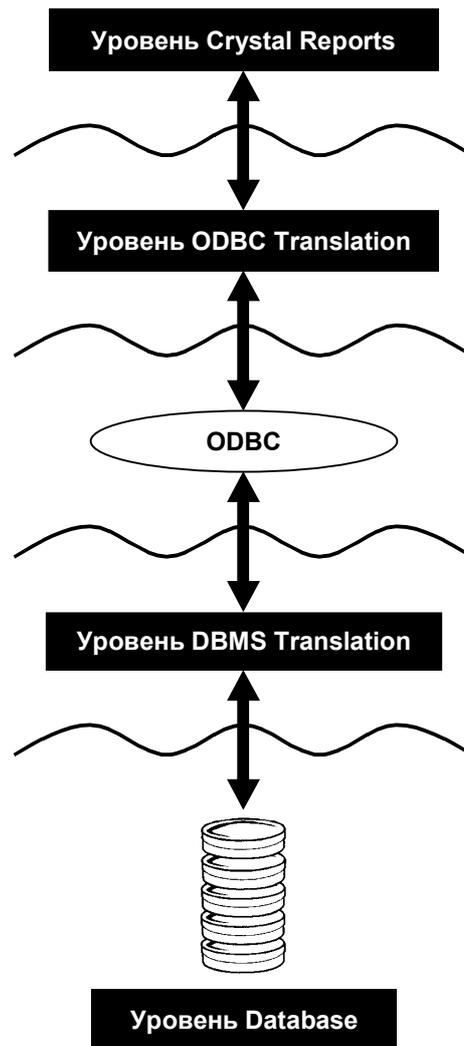
Кроме того, источники данных ODBC до того, как их можно будет использовать, должны быть сконфигурированы и корректно установлены в файлах ODBC.INI и ODBCINST.INI. Если Вы создаете отчет в одной системе и пытаетесь открыть его в другой системе, для которой установлен другой источник данных ODBC, Crystal Report Designer не сможет связаться с данными.

При работе с ODBC Вы должны помнить, что используемый ODBC язык SQL основан на стандартах, установленных для языка SQL Американским национальным институтом стандартов (ANSI). Однако некоторые приложения DBMS на основе SQL поддерживают дополнительные свойства языка SQL, специфические для данной DBMS. Если Вы используете свойства, уникальные для Вашей DBMS, то ODBC не сможет их транслировать (хотя в большинстве случаев он все-таки найдет большую часть данных). Смотрите *Язык SQL*.

Пять уровней

Процесс, посредством которого Crystal Report Designer получает доступ к данным из источника данных ODBC, состоит из пяти уровней:

1. Уровень Crystal Report Designer.
2. Уровень ODBC Translation.
3. Уровень ODBC.
4. Уровень DBMS Translation (источник данных ODBC).
5. Уровень Database.



Используя для связи Язык структурированных запросов (SQL), все пять уровней могут в удобной форме передавать данные из базы данных в отчет.

Уровень Crystal Report Designer

При работе с данными ODBC Crystal Report Designer генерирует оператор SQL, запрашивающий соответствующие данные у ODBC. Мощный генератор SQL, встроенный в Crystal, разработан для создания оператора SQL, позволяющего переложить максимально возможную часть работы по генерации отчета на источник данных ODBC или сам ODBC и возвращающего только те данные, которые необходимы для создания отчета. Любое лицо, знакомое с SQL, может просматривать и настраивать этот оператор SQL. Смотрите *How to edit an SQL query*.

Уровень ODBC Translation

Для осуществления связи с ODBC Crystal Report Designer использует Библиотеку динамической связи PDSODBC.DLL. Этот файл уникален для среды Crystal Report Designer и обеспечивает Вашему отчету доступ к любому источнику данных ODBC. Он представляет собой драйвер, который фактически передает данные в ODBC и обратно.

Уровень ODBC

ODBC представляет собой набор из нескольких DLL и файлов INI, встроенных в среду Windows, выполняющий роль шлюза, через который могут передаваться запросы базы данных и данные. Любой файл или формат базы данных, используемый через ODBC, должен быть установлен как источник данных ODBC.

В 16-битовой версии Windows (Windows 3.x) ODBC составляют следующие первичные Библиотеки динамической связи:

- ODBC.DLL,
- ODBCINST.DLL.

Информация об источнике данных ODBC хранится в:

- ODBC.INI,
- ODBCINST.INI.

ODBC использует эти файлы для получения и использования информации об источнике данных.

В 32-битовой версии Windows (Windows 95 и Windows NT) в ODBC входят следующие библиотеки:

- ODBC32.DLL (32-битовая версия ODBC.DLL),
- ODBCCP32.DLL (32-битовая версия ODBCINST.DLL)
- ODBCINT.DLL (ODBC 2.5 и более поздние версии).

Хотя информация, относящаяся к источнику данных, остается записанной в ODBC.INI и ODBCINST.INI, 32-битовый ODBC использует базу данных Windows Registry для поиска информации об отдельных источниках данных.

ВНИМАНИЕ: Для получения более полной информации об ODBC и файлах ODBC смотрите документацию ODBC Microsoft.

ODBC использует язык SQL для осуществления всех транзакций между Crystal Report Designer и ODBC. Даже если база данных обычно не использует SQL для создания и работы с таблицами, драйвер ODBC, поддерживаемый базой данных (уровень DBMS Translation), должен осуществлять связь с ODBC посредством SQL. Для большинства пользователей данное свойство ODBC прозрачно, но некоторые наиболее опытные пользователи часто пользуются преимуществами свойств языка SQL, используемого ODBC.

Уровень DBMS Translation (источник данных ODBC)

Данный уровень состоит из одного или нескольких драйверов, поддерживаемых DBMS, которые позволяют ODBC осуществлять связь с базой данных. Crystal Report Designer поставляется с несколькими драйверами ODBC для многих из наиболее популярных форматов баз данных. Если Вы не уверены в том, можете ли Вы использовать драйвер ODBC для доступа к данным в Вашей базе данных, обратитесь к документации для своего приложения DBMS. Большинство приложений, выполняемых на платформе Windows, предлагают драйвер ODBC.

Если DBMS поддерживает драйвер ODBC, то она должна зарегистрировать драйвер с ODBC на машине, на которой он установлен. Это осуществляется путем задания драйверу имени и

записи имени файла в файл ODBCINST.INI. Обычно данный шаг выполняется автоматически при инсталляции DBMS в систему. Однако Ваша сетевая система или приложение DBMS могут потребовать, чтобы Вы зарегистрировали драйвер ODBC вручную, используя приложение Setup ODBC. Для получения более полной информации о регистрации драйвера ODBC с ODBC обратитесь к документации для своей DBMS.

После того, как драйвер ODBC зарегистрирован, Вам необходимо установить с помощью этого драйвера Источник данных ODBC. Источник данных ODBC представляет собой объект, к которому Вы присоединяетесь при осуществлении доступа к данным из Crystal Report Designer через ODBC. Источники данных записаны в файле ODBC.INI. Источник данных отслеживает трансляционные файлы DBMS (драйверы ODBC) и, иногда, саму базу данных. Источник данных ODBC может задавать только формат базы данных, такой как Oracle, Gupta, Сервер Sybase SQL или MS SQL. Однако, некоторые пользователи предпочитают фактически задавать определенную базу данных. В этом случае, источник данных ODBC расширяется и охватывает как уровень ODBC Translation, так и уровень Database.

Если Вы используете базу данных клиент/сервер, такую как сервер SQL, то драйверы ODBC осуществляют связь с сервером базы данных через Коммуникационный уровень базы данных, т.е. тот же уровень, который клиент базы данных использует для связи с сервером базы данных.

ВНИМАНИЕ: Драйверы ODBC находят свои специфические клиентские файлы DBMS на локальной машине вручную посредством ключевых директорий, которые инсталлированы клиентом DBMS в поисковом пути (заданном в AUTOEXEC.BAT). Важным моментом является то, что клиентская рабочая станция на локальном PC должна иметь возможность успешно связаться со своим сервером. Если Вы не знаете, как это можно проверить, свяжитесь со своим менеджером IS.

Для драйверов ODBC и Crystal Report Designer не имеет значения, на какой платформе работает Ваш сервер базы данных, т.к. присоединяется к серверу и осуществляет с ним связь клиент DBMS; Crystal Report Designer и драйверам необходимо связываться только с клиентом DBMS. Смотрите *Using SQL and SQL databases*.

Уровень Database

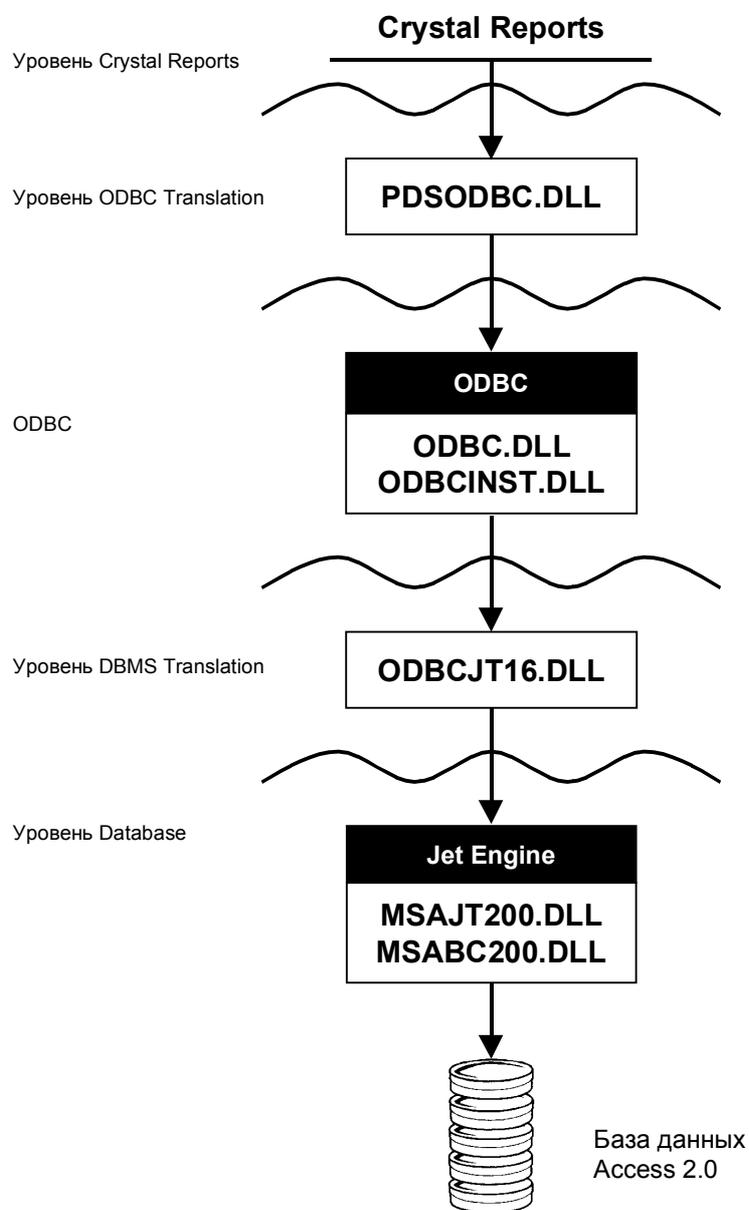
Файл базы данных, к которому обращается источник данных ODBC, может находиться в любом месте системы. После того, как источник данных ODBC установлен, Crystal Report Designer нет необходимости знать фактическое местоположение и формат данных. Следовательно, база данных может иметь любой формат и может располагаться в любом месте сети до тех пор, пока ODBC может осуществлять с ней связь через источник данных ODBC.

Обычные форматы базы данных ODBC

Access

Хотя файлы Microsoft Access можно открывать непосредственно с помощью Crystal Report Designer, Вы, возможно, захотите использовать для этого ODBC. При установлении связи с базами данных Access ODBC использует трансляционный файл, который связывается с Механизмом базы данных Jet Microsoft. Механизм базы данных Jet представляет собой пример драйвера ODBC, специфического для DBMS.

Механизм Jet является частью Access DBMS и осуществляет всю фактическую работу с базой данных Access. Это необходимый компонент для работы с данными Access. Так как механизм Jet является неотъемлемой частью всех баз данных Access, на приведенном ниже рисунке он представлен как часть уровня Database.



ВНИМАНИЕ: На приведенной выше диаграмме представлены файлы, необходимые Crystal Report Designer для работы с базами данных, используемыми 16-битовую версию Microsoft Access 7.0. Если Вы пользуетесь другой версией Access, то для получения информации о файлах, необходимых для открытия данных Access, обратитесь к оперативной Подсказке Runtime File Requirements (RUNTIME.HLP).

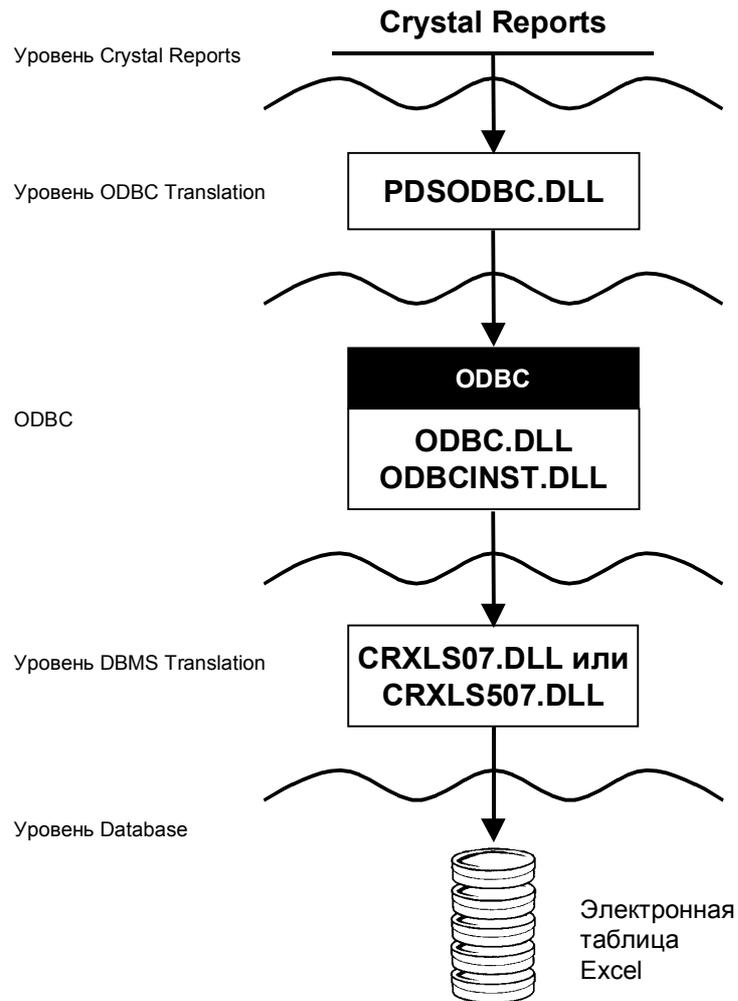
Excel

Электронные таблицы Microsoft Excel могут быть конвертированы в базы данных, которые Crystal Report Designer может читать с помощью ODBC. В Excel 4.0 и более ранних версиях используйте команду **Set Database** из меню Data. В Excel 5.0 и более поздних версиях используйте команду **Define** из меню Insert | Name. После конвертации строки электронных таблиц становятся записями, а колонки – полями. (Для получения более полной информации о конвертации электронных таблиц в формат базы данных обратитесь к документации по Excel.) После того, как электронная таблица конвертирована, Вы можете установить источник данных ODBC для файла, а затем выбрать его из Crystal Report Designer.

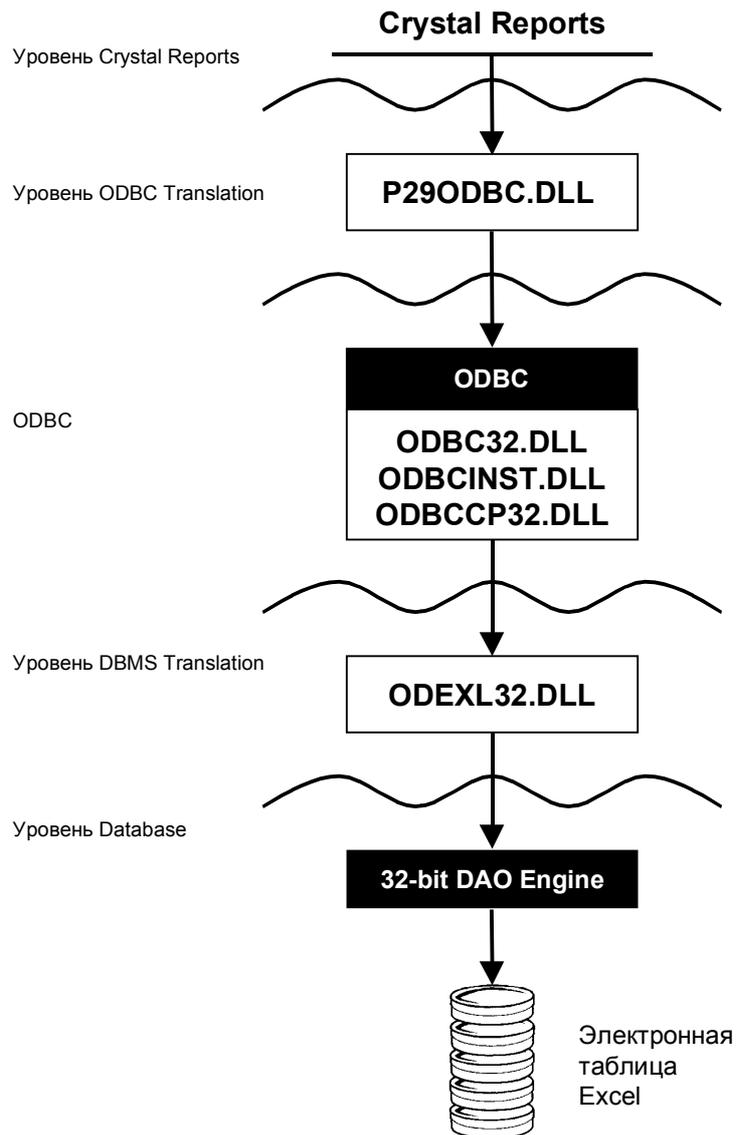
ВНИМАНИЕ: Если Вы используете Excel 7.0 или более позднюю версию, то Вы можете экспортировать свои электронные таблицы как таблицы базы данных Access, а затем считывать их из Crystal Report Designer, как Вы делаете это с другими таблицами Access. Дополнительную информацию смотрите в документации по Excel.

Базы данных Excel для 16-битовой среды Windows считываются через трансляционный файл CRXLS07.DLL (для Excel 4.0 и более ранних версий) или через файл CRXLS507.DLL (для Excel 5).

Crystal Report Designer устанавливает следующие драйверы. ODBC может связываться с таким драйвер для прочтения конвертированной электронной таблицы Excel.



Если Вы используете 32-битовую систему Windows, то можете установить источник данных ODBC для 32-битовых электронных таблиц Excel с помощью 32-битового механизма DAO. Механизм DAO устанавливается в Вашу систему при инсталляции 32-битовой версии программы. Однако, для электронной таблицы Excel Вы должны будете установить источник данных ODBC вручную. Смотрите *Как устанавливать источник данных ODBC*.



Базы данных SQL через INTERSOLV DataDirect

Существует множество различных систем управления базами данных SQL, и доступ к большинству из них можно осуществлять через ODBC. Crystal Report Designer автоматически устанавливает и устанавливает несколько источников данных ODBC, обеспечивая Вам доступ к большинству популярных баз данных SQL, включая:

- Oracle
- Sybase SQL Server
- MS SQL Server
- Gupta SQLBase
- Scalable SQL
- DB2/2
- ASCII Text

ВНИМАНИЕ: ASCII Text не является в действительности форматом базы данных SQL, но представляет собой текстовые файлы, которые программа Crystal Report Designer может считывать, используя ODBC, так же, как и многие другие базы данных SQL.

Для этих форматов баз данных SQL Crystal Report Designer поддерживает драйверы ODBC INTERSOLV DataDirect. Посредством этих драйверов можно получить доступ не ко всем базам данных SQL, но, для удобства, программа автоматически устанавливает их во время инсталляции. Crystal Report Designer, как обычно, осуществляет связь с ODBC с помощью трансляционного файла PDSODBC.DLL, но драйверы DataDirect упрощают доступ ODBC к фактическим базам данных.

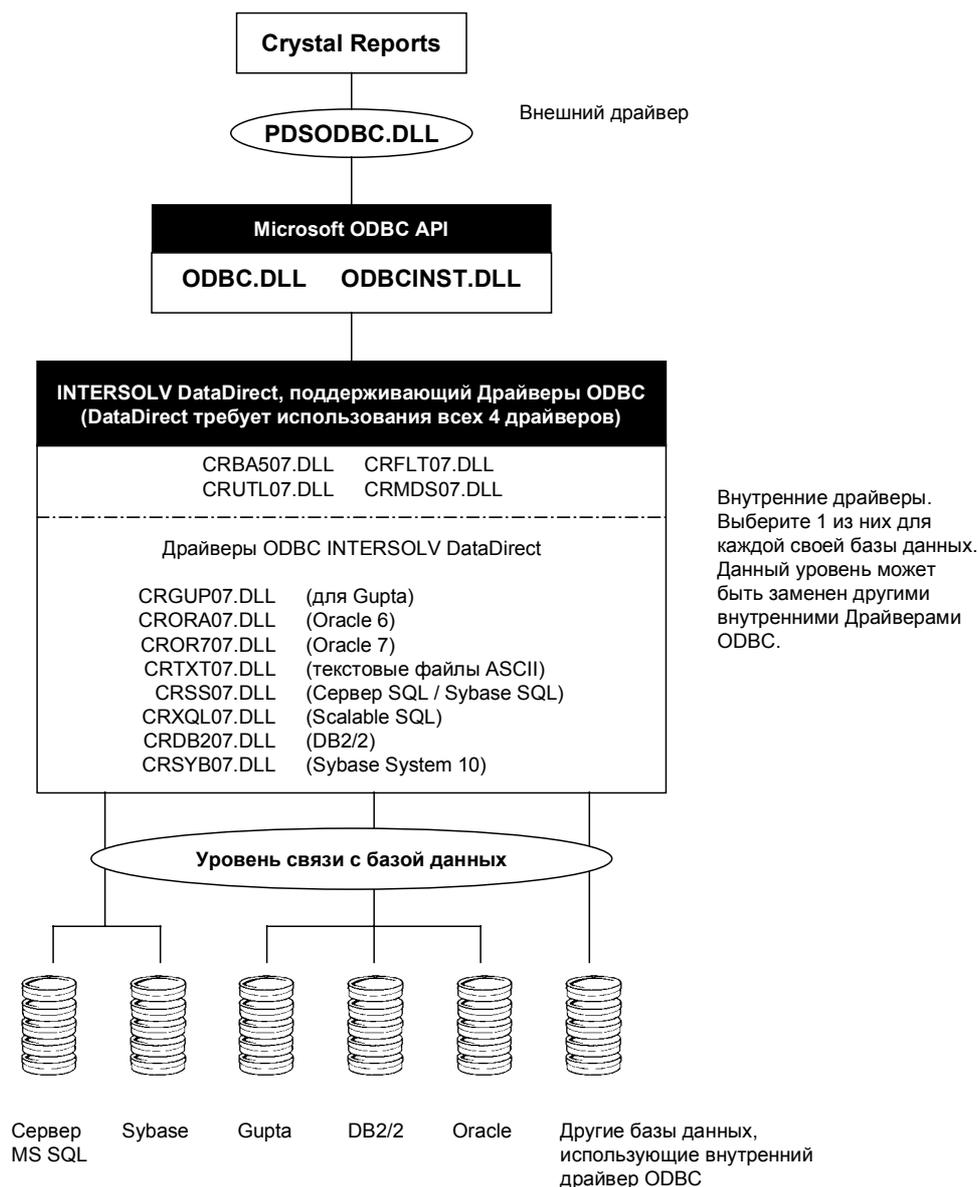
ВНИМАНИЕ: Если Вы используете базу данных SQL, недоступную для драйверов INTERSOLV, то более полную информацию о необходимых Вам драйверах ODBC Вы можете получить из документации к своей DBMS.

Не путайте базы данных SQL с языком SQL. Язык структурированных запросов (SQL) является специальным компьютерным языком, используемым для разработки, построения и чтения файлов баз данных. Смотрите *Использование SQL и баз данных SQL*.

Базы данных SQL представляют собой любой набор компьютерных приложений, когда для создания и работы с базой данных требуется язык SQL. Приложения SQL DBMS создаются, обычно, для выполнения больших приложений клиент/сервер по сети.

Чтобы открыть базу данных SQL, ODBC использует набор из четырех драйверов поддержки ODBC DataDirect (одинаковые для всех форматов баз данных SQL), драйвер ODBC DataDirect для определенного формата базы данных и Уровень связи с базой данных, который осуществляет непосредственную связь с файлом базы данных. Все файлы, поддерживаемые для данного трансляционного уровня DBMS, представлены драйверами INTERSOLV DataDirect.

ВНИМАНИЕ: ODBC не приходится использовать драйверы INTERSOLV DataDirect для приводимых здесь баз данных SQL. Большинство приложений DBMS SQL поддерживает собственные драйверы ODBC. Однако Crystal Report Designer автоматически устанавливает драйверы INTERSOLV.



ВНИМАНИЕ: 32-разрядная версия Seagate Crystal Report поддерживает 32-битовые драйверы INTERSQLV для доступа к базам данных SQL. Эти 32-битовые драйверы называются CR*09.DLL вместо CR*07.DLL.

Как было упомянуто ранее, Crystal Report Designer автоматически создает несколько источников данных ODBC для форматов баз данных SQL, поддерживаемых INTERSQLV DataDirect.

Эти источники данных отображаются в диалоговом окне Log On Server:

- ODBC-CRDBM поддерживает DB2/2
- ODBC-CRGUP поддерживает Gupta
- ODBC-CROR7 поддерживает Oracle 7
- ODBC-CRORA поддерживает Oracle 6
- ODBC-CRSS поддерживает серверы MS SQL и Sybase SQL
- ODBC-CRXQL поддерживает Scalable SQL
- ODBC-CRSYB поддерживает Sybase System 10.

Библиотека INTERSOLV DataDirect

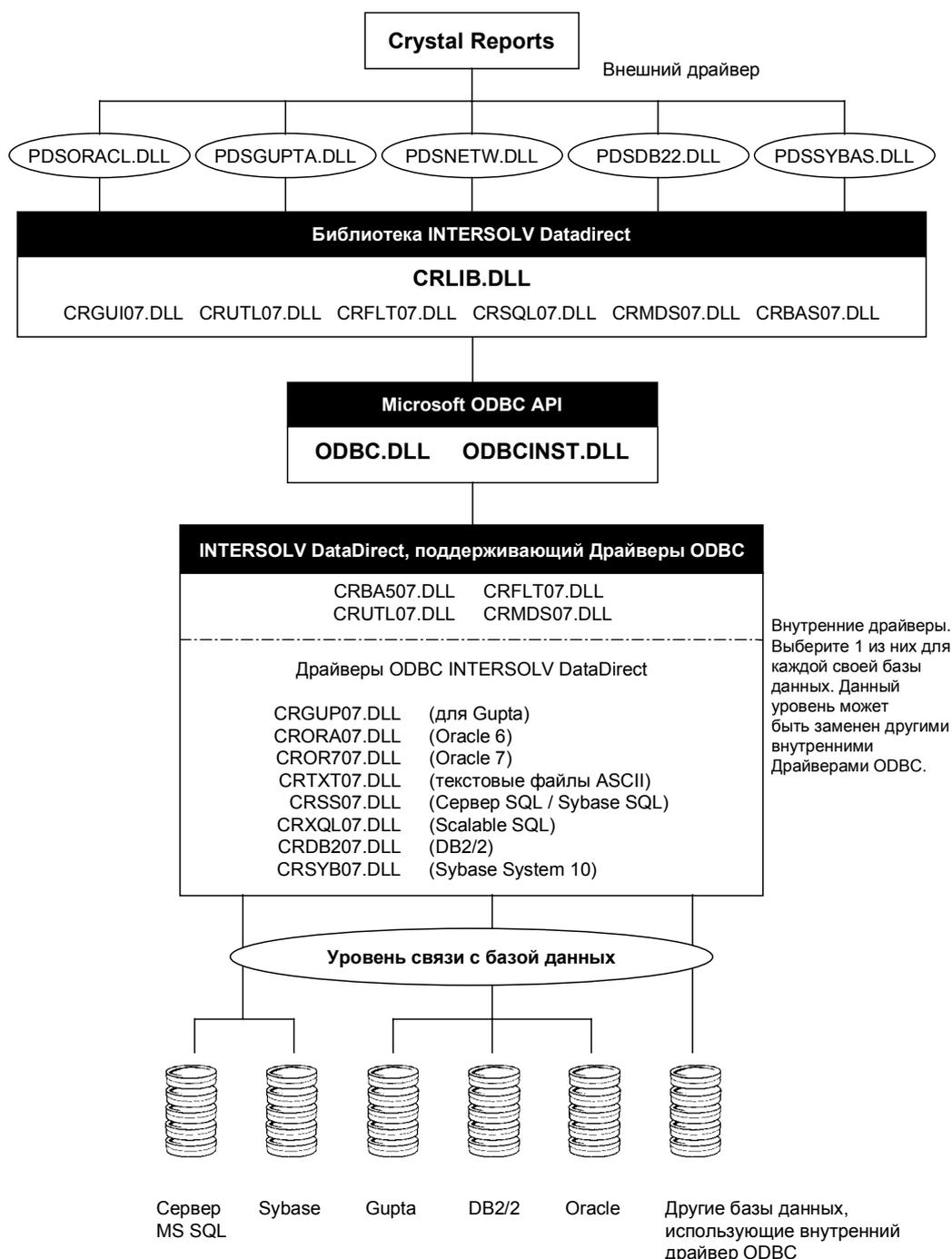
Вместо осуществления прямой связи с ODBC посредством файла PDSODBC.DLL, Crystal Report Designer может использовать Библиотеку INTERSOLV DataDirect, поставляемую с Crystal Report Designer. Эти интерфейсные драйверы используют стандарт ODBC в качестве базового уровня. Однако, вместо использования PDSODBC.DLL для связи с ODBC, Crystal Report Designer использует специальный файл PDS*.DLL, соответствующий базе данных SQL, к которой осуществляется доступ.

Сравнение Библиотеки DataDirect и ODBC

Преимущество использования Библиотеки DataDirect по сравнению с прямым ODBC заключается в возможности Библиотеки передавать частный синтаксис SQL на определенные серверы. Например, т.к. Oracle имеет собственный драйвер PDS, разработчики SQL, работающие с данными Oracle, смогут периодически передавать на сервер более точные операторы SQL. Таким образом, Crystal Report Designer сможет передавать уникальные для определенной DBMS свойства языка SQL. При прямом использовании ODBC PDSODBC.DLL является обычной DLL, используемой для всех типов DBMS, а использование частного синтаксиса SQL на базе определенных серверов может оказаться практически невозможным.

Структура Библиотеки DataDirect

С библиотекой INTERSOLV DataDirect Crystal Report Designer использует транслятор ODBC, специфический для базы данных, к которой осуществляется доступ. Этот файл, в свою очередь, связывается с Библиотекой DataDirect. Библиотека DataDirect осуществляет окончательную трансляцию оператора SQL в ODBC.



ВНИМАНИЕ: 32-разрядная версия Seagate Crystal Report поддерживает 32-битовые драйверы INTERSOLV для доступа к базам данных SQL. Эти 32-битовые драйверы называются CR*09.DLL вместо CR*07.DLL.

Базы данных Библиотеки INTERSOLV DataDirect

- Oracle
- Sybase SQL Server
- MS SQL Server
- Gupta SQLBase
- Scalable SQL
- DB2/2

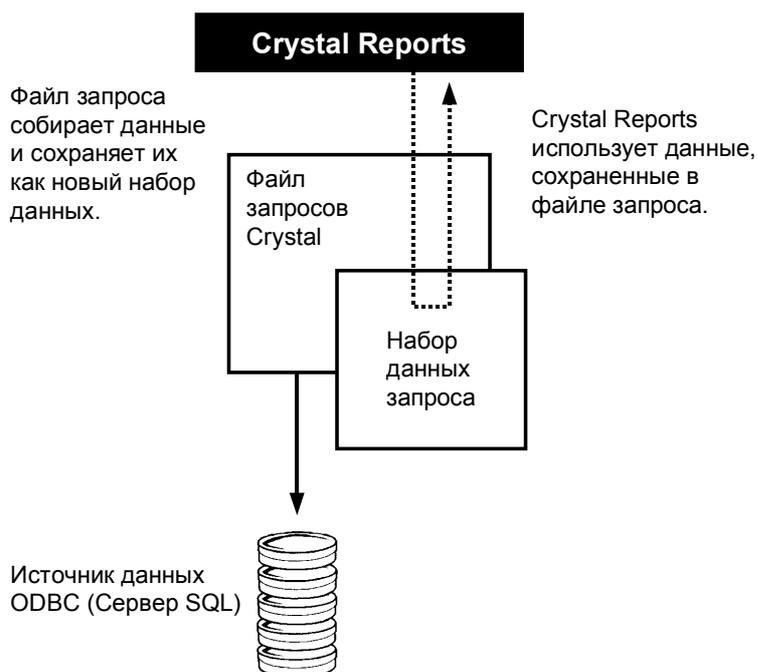
ВНИМАНИЕ: Источники данных ODBC, которые Crystal Report Designer устанавливает для Библиотеки INTERSOLV DataDirect, отображены в диалоговом окне Log On Server без префикса "ODBC-", используемого для источников данных, идущих через прямой ODBC.

Файлы Crystal Query Designer

Query (запрос) представляет собой запрос об определенном наборе данных из базы данных. После того, как данные собраны, они могут сохраняться как файл Crystal Query Designer (.QRY). Затем файл Crystal Query Designer может использоваться подобно таблице базы данных. Данные, собранные в него из Ваших баз данных SQL, становятся новым набором данных, который можно добавлять в отчеты.

Файл Crystal Query Designer собирает данные из источников данных ODBC, таких как серверы SQL, с помощью SQL (Язык структурированных запросов). Оператор SQL с инструкцией собрать и вернуть определенный набор данных посылается непосредственно в SQL DBMS. DBMS проводит сбор, сортировку и группирование всех фактических данных в соответствии с инструкциями в операторе SQL. Только окончательный набор данных возвращается в Query Designer и сохраняется как файл запроса.

ВНИМАНИЕ: Если Вы не знакомы с SQL, смотрите Типы соединений SQL (источники данных ODBC).



Сначала Вы уточняете оператор SQL и создаете файл запроса, содержащий только необходимые Вам данные. Затем Вы создаете свои отчеты на основе файла запроса, а не оригинальных баз данных. Большая часть процесса по сбору данных осуществляется на сервере SQL, когда Вы создаете запрос. При создании Вами отчета Crystal Report Designer приходится работать только с небольшим набором данных, что сберегает Ваше время и усилия. Смотрите *How to create a new query*.

Преимущества

Хотя существует множество доводов в пользу использования запросов, главным из них считается возможность переложить большую часть работы по поиску данных на сервер и сохранить полученные данные в качестве отдельного набора данных. Используя технологию передачи SQL, когда решение задач по поиску данных перелagается на сервер, Вы сможете более эффективно и быстро находить нужные Вам данные. Всегда при работе с данными SQL Вы должны помнить о возможности сначала создать файл запроса, а потом уже создавать свой отчет.

Кроме того, существует Query Expert, обеспечивающий большее управление Вашими данными со стороны языка SQL. Хотя Crystal Report Designer поддерживает некоторые свойства языка SQL при доступе к данным ODBC или SQL, Query Expert является мощным инструментом запросов, поддерживающим более мощные свойства доступа к данным SQL. Query Expert позволяет использовать функции суммирования в операторе SQL и поддерживает все элементы запросов данных, имеющиеся в языке SQL. Если Вы знаете язык SQL, то сможете выполнять комплексные задачи SQL с использованием операций UNION и подзапросов. Смотрите *Язык SQL*.

Недостатки

Так как запрос представляет собой законченный набор данных, любые записи, поля или таблицы, не включенные в запрос при его создании, становятся недоступными, когда Вы используете запрос для построения отчета. Отчет не может быть сложнее тех данных, к которым он осуществляет доступ, поэтому, при использовании запроса, Вы ограничиваете возможности для создания отчета теми данными, которые присутствуют в запросе.

Кроме того, запрос не может использоваться с любым источником данных. Если Вы решаете использовать запрос в своем отчете, то уже не сможете использовать в отчете дополнительный источник данных, такой как таблица базы данных или другой файл запроса.

Словарные (Dictionary) файлы Crystal

Словарный файл (.DC5) представляет собой структурированный и упрощенный вид организационных данных, который Вы можете создавать для некоторых или всех лиц в Вашей организации, использующих Crystal Report Designer. При использовании словаря конечные пользователи видят только необходимый им поднабор таблиц и полей.

ВНИМАНИЕ: Приложение по созданию словарей поставляется только с Профессиональной редакцией Crystal Report Designer. Данное приложение предназначено для Менеджеров IS и баз данных, которые отвечают за предоставление пользователям вариантов доступа к данным. Как Стандартная, так и Профессиональная редакции Crystal Report Designer могут открывать и использовать словарные файлы.

Словарь является необязательным источником данных для Crystal Report Designer. Он является удобным фильтром, делающим сложные технологии доступа к данным более простыми и ясными для конечных пользователей, но, при этом, пользователи могут также осуществлять прямой доступ к данным из базы данных. Словари просто предоставляют удобства, не налагая, при этом, никаких ограничений.

Однако, если при построении отчета Вы решите использовать словарь, то в качестве источника данных для своего отчета Вы сможете использовать только данный словарь. Другими словами, Вы не сможете одновременно открыть Словарь и источник данных ODBC, базу данных с прямым доступом или какой-либо другой источник данных из одного отчета. Для этого потребуются другие словари. Назначением словаря является обеспечение быстрого и простого для понимания доступа к определенному набору данных. Это несовместимо с присоединением других источников данных в том же отчете. Смотрите *How to create a new dictionary*.

Словари позволяют Вам:

- создать один динамический вид всех данных, необходимых для создания организационных отчетов и запросов;
- создать словарь, содержащий множество источников данных, таблиц и связей;
- организовать данные и переименовать таблицы и поля так, чтобы пользователям было легче понять содержание и назначение данных;
- создать комплексные формулы работы с данными, которые пользователи смогут применять, не вдаваясь в понимание концепций этих формул.

Словари уменьшают денежные и временные затраты на поддержку, увеличивают производительность работы пользователей, а также снижают вероятность неправильного использования, потери и разрушения данных.

Созданный словарь действует как фильтр, обеспечивая четкий и ясный для понимания конечного пользователя вид сложных данных. При его применении пользователи смогут создавать прекрасные отчеты даже на основе сложных данных.



Crystal Report Designer читает словарный файл с помощью файла PDIXBSE.DLL (для 32-разрядной версии – P2IXBSE.DLL). Данный трансляционный файл базируется на механизме xBase, используемом для доступа к базам данных *dBASE*, *FoxPro*, *Clipper*, и обеспечивает гибкость, необходимую для чтения словарей.

ВНИМАНИЕ: Если Вы обновили более раннюю версию *Crystal Report Designer*, допуская использование словарных файлов, то необходимо иметь в виду, что старые словарные файлы (.DCT) базировались на механизме *Vtrieve*. Если Вы конвертируете эти старые файлы в новые словарные файлы, то новые файлы будут использовать вместо *Vtrieve* трансляционный уровень *xBase*. Однако механизм *Vtrieve*, инсталлированный старой версией *Crystal Report Designer*, должен продолжать оставаться в Вашей системе для того, чтобы новое словарное приложение могло считывать старые словарные файлы.

Уровень доступа к базе данных (Database Access Layer) представляет собой уровень, через который словарный файл считывает данные из оригинальной базы данных. Словари должны считывать данные из базы данных, используя те же пути, что и *Crystal Report Designer*. В приведенных в данной главе разделах об уровнях баз данных с прямым доступом и источниках данных ODBC указаны файлы, необходимые словарям для открытия данных.

Преимущество словарей заключается в том, что этот Уровень доступа к базам данных является абсолютно прозрачным для пользователя. Словарь отображает данные в *Crystal Report Designer* в таком виде, который может соответствовать, а может и не соответствовать фактическим данным, но в этом виде пользователям будет легко осуществлять доступ и использовать данные.

И, наконец, словари поддерживают простые средства для обмена оригинальными данными без изменения их внешнего вида. Например, Ваша оригинальная база данных может содержать поля *Price*, *Quantity* и *Cost*, а Вы решите, что *Cost* можно вычислить из *Price* и *Quantity*; в результате Вы удаляете поле *Cost*. Путем простого обновления словаря так, чтобы значения поля *Cost* вычислялись, а не брались непосредственно из базы данных, Вы исключаете поле *Cost*, и Ваши пользователи никогда не заметят этого изменения. Это также очень полезно при изменении формата всей оригинальной базы данных. И в этом случае Вам также необходимо будет только обновить словарь.

Часто словарные файлы создаются менеджером IS, который ведет базу данных компании. Менеджер IS может работать с организованными и классифицированными данными, создавая несложные, простые в применении словари для служащих компании. Благодаря тому, что оригинальные данные были обработаны и настроены менеджером IS с учетом возможности их применения в самых различных целях, в каждом словарном файле служащие будут находить только необходимые им данные.