

ПРАКТИКУМ [P.002]

Базовые навыки работы в Deductor Studio

Занятие 1. Общие сведения

Развитие и назначение Deductor

Deductor – это аналитическая платформа, основа для создания законченных прикладных решений в области анализа данных. Реализованные в Deductor технологии позволяют на базе единой архитектуры пройти все этапы построения аналитической системы: от консолидации данных до построения моделей и визуализации полученных результатов.

До появления аналитических платформ анализ данных осуществлялся в основном в статистических пакетах. Их использование требовало высокой квалификации пользователя. Большинство алгоритмов, реализованных в статистических пакетах, не позволяло эффективно обрабатывать большие объемы информации. Для автоматизации рутинных операций приходилось использовать встроенные языки программирования.

В конце 80-х гг. произошел стремительный рост объемов информации, накапливаемый на машинных носителях и возросли потребности бизнеса по применению анализа данных. Ответом этому стало появление новых парадигм в анализе: хранилища данных, машинное обучение, Data Mining, Knowledge Discovery in Databases. Это позволило популяризировать анализ данных, вывести его на промышленную основу и решить огромное число бизнес-задач с большим экономическим эффектом.

Венцом развития анализа данных стали специализированные программные системы – аналитические платформы, которые полностью автоматизировали все этапы анализа от консолидации данных до эксплуатации моделей и интерпретации результатов.

Первая версия Deductor увидела свет в 2000 г. и с тех пор идет непрерывное развитие платформы. В 2007 г. выпущена пятая по счету версия системы, в 2009 г. – версия 5.2.

Сегодня Deductor – это яркий представитель как настольной, так и корпоративной системы анализа данных последнего поколения.

Общие сведения о Deductor

Аналитическая платформа Deductor состоит из пяти частей:

- **Warehouse** – хранилище данных, консолидирующее информацию из разных источников;
- **Studio** – приложение, позволяющее пройти все этапы построения прикладного решения, рабочее место аналитика;
- **Viewer** – рабочее место конечного пользователя, одно из средств тиражирования знаний (т.е. когда построенные аналитиком модели используют пользователи, не владеющие технологиями анализа данных);
- **Server** – служба, обеспечивающая удаленную аналитическую обработку данных;
- **Client** – клиент доступа к Deductor Server. Обеспечивает доступ к серверу из сторонних приложений и управление его работой.

Существует три типа варианта поставки платформы Deductor:

- Enterprise;
- Professional;
- Academic.

В зависимости от типа поставки набор доступных компонентов может различаться.

Версия **Enterprise** предназначена для корпоративного использования. В ней присутствуют:

- Серверные компоненты Deductor Server и Deductor Client.
- Интерфейс доступа к Deductor через механизм OLE Automation.
- Традиционное хранилище данных **Deductor Warehouse** на трех СУБД: Firebird, MS SQL, Oracle.
- Виртуальное хранилище данных **Deductor Virtual Warehouse**.

Версия **Professional** предназначена для небольших компаний и однопользовательской работы. В ней отсутствуют серверные компоненты, поддержка OLE, виртуальное хранилище, а традиционное хранилище данных можно создавать только на СУБД FireBird. Автоматизация выполнения сценариев обработки данных осуществляется только через пакетный режим.

Версии **Professional** и **Enterprise** требуют установки драйверов Guardant для работы с лицензионным ключом.

Версия **Academic** предназначена для образовательных и обучающих целей. Ее функционал аналогичен версии **Professional** за исключением:

- отсутствует пакетный запуск сценариев, т.е. работа в программе может вестись только в *интерактивном режиме*;
- отсутствует импорт из промышленных источников данных: 1С, СУБД, файлы MS Excel, Deductor Data File;
- некоторые другие возможности.

Категории пользователей Deductor

В процессе развертывания и использования аналитической платформы с ней взаимодействуют различные категории пользователей. Можно выделить четыре основные категории:

- аналитик;
- пользователь;
- администратор;
- программист.

Функции аналитика:

- создание в Deductor Studio сценариев – последовательности шагов, которую необходимо провести для получения нужного результата.
- построение, оценка и интерпретация моделей.
- настройка панели отчетов для пользователей Deductor Viewer.
- настройка сценария на поточную обработку новых данных.

Функции пользователя:

- просмотр готовых отчетов в Deductor Viewer.

Функции администратора:

- установка компонентов Deductor на рабочих местах и сервера ключей Guardant при необходимости.
- развертывание традиционного хранилища данных на сервере.
- контроль процедур регулярного пополнения хранилища данных.

- конфигурирование сервера Deductor Server.
- настройка пакетной и/или серверной обработки сценариев Deductor.
- оптимизация доступа к источникам данных, в том числе к хранилищу данных.

Функции программиста:

- интеграция Deductor с источниками и приемниками данных.
- вызов Deductor из внешних программ различными способами, в том числе взаимодействие с Deductor Server.

Такая работа как проектирование и наполнение хранилища данных часто выполняется коллективно аналитиком, администратором и программистом. Аналитик проектирует семантический слой хранилища данных, то есть определяет, *какие* данные необходимо иметь в хранилище. Администратор создает хранилище данных и наполняет его данными. Программист при необходимости создает программные модули, выполняющие выгрузку информации из учетных систем в промежуточные источники (так называемые *транспортные таблицы*).

Установка Deductor

Установку Deductor рекомендуется проводить администратору системы, однако, при наличии прав администратора в Windows это может сделать и аналитик. Установка может быть произведена на компьютер с операционной системой MS Windows 2000 и выше. Системные требования к компьютеру изложены в справочной системе.

Для установки Deductor Professional/Academic запустите файл инсталлятора и следуйте инструкциям по установке. На странице **Выбор компонентов** программы установки предоставляется выбор, какой набор компонентов пакета Deductor необходимо установить на компьютер. В выпадающем списке можно выбрать predetermined конфигурации установки платформы, и программа установки сама предложит нужный набор компонентов.

После установки программ серии **Professional** и **Enterprise** дополнительно потребуется настроить работу с электронным ключом защиты от копирования. Установку и подключение электронного ключа осуществляет администратор.

Существуют два вида ключей – локальный и сетевой. Локальный ключ устанавливается на том же компьютере, что и Deductor, и работать с ним можно только с этой рабочей станцией. Сетевой ключ устанавливается на сервере, и к нему могут подключаться несколько пользователей одновременно (количество пользователей ограничивается типом приобретаемой лицензии).

При каждом запуске Deductor пытается найти доступный электронный ключ. В случае если ключ не найден, могут появиться следующие сообщения об ошибке:





При наличии таких ошибок следует обратиться к администратору.

Практическая работа:

- 1 Установите Deductor (конфигурация Deductor Studio – рабочее место аналитика) и убедитесь, что он запускается.

Вопросы для проверки:

- 1 Из каких частей состоит Deductor?
- 2 Какие варианты поставки Deductor существуют?
- 3 Чем отличается версия **Professional** от **Academic**?
- 4 Имеются ли ограничения по количеству обрабатываемых записей в версии Deductor Academic?
- 5 Сколько категорий пользователей Deductor можно выделить?
- 6 Перечислите функции аналитика.
- 7 Кто обычно занимается проектированием и наполнением хранилища данных?
- 8 Каким образом лицензируется Deductor?
- 9 У вас установлен Deductor. При его запуске появляется сообщение об ошибке:

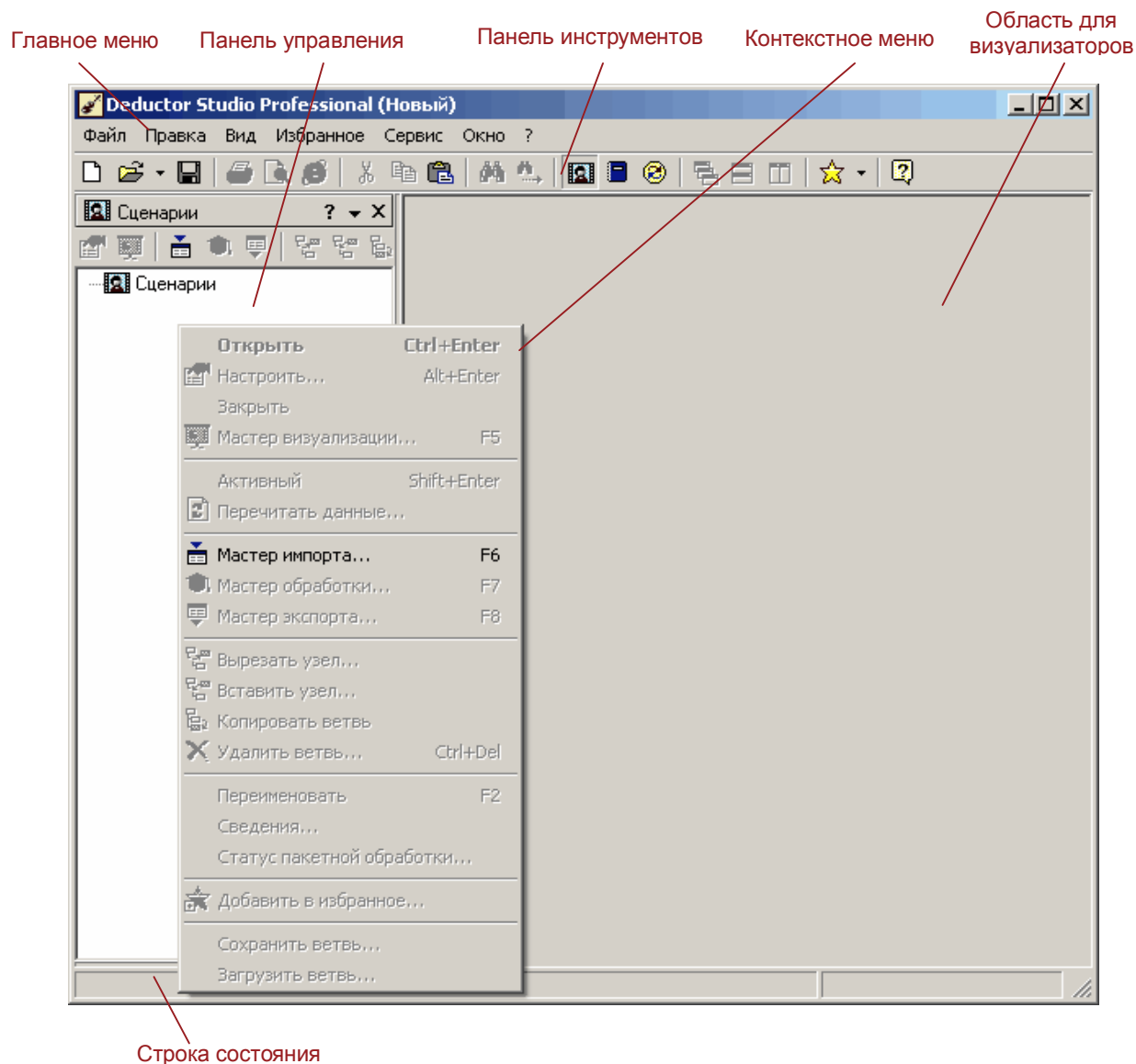
Windows NT driver is required.

Какова наиболее вероятная причина ошибки?



Занятие 2. Начало работы с системой

Главное окно Deductor Studio

После запуска главное окно Deductor Studio выглядит следующим образом.

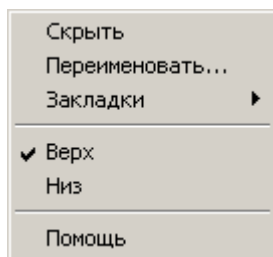


По умолчанию панель управления представлена одной вкладкой **Сценарии**. Кроме того, доступны еще две вкладки: **Отчеты** и **Подключения**. Сделать их видимыми можно следующими способами:

- главное меню **Вид ► Отчеты** и **Вид ► Подключения**
- кнопки  и  на панели инструментов.

Можно производить «drag & drop» манипуляции с вкладками, меняя их расположение и порядок.


При нажатии правой кнопки мыши на любой вкладке появляется контекстное меню:



- Скрыть – делает вкладку невидимой;
- Переименовать – переименовывает название вкладки;
- Закладки – переключается на выбранную закладку;
- Верх/Низ – задает расположение названий вкладок: вверху либо внизу;
- Помощь – открывает раздел справки.

Справка по программе

Справка по программе вызывается из главного окна системы следующими способами:

- главное меню ? ► **Справка**,
- клавиша **F1**,
- кнопка на панели инструментов .

Помощь содержит подробное описание работы с Deductor Studio: системные требования, настройки узлов, способы осуществления действий с объектами системы.

Понятие проекта

В Deductor Studio ключевым понятием является *проект*. Это файл с расширением ***.ded**, по структуре соответствующий стандартному xml-файлу. Он хранит в себе:

- последовательности обработки данных (сценарии);
- настроенные визуализаторы;
- переменные проекта и служебную информацию.

Пример фрагмента файла ***.ded**:

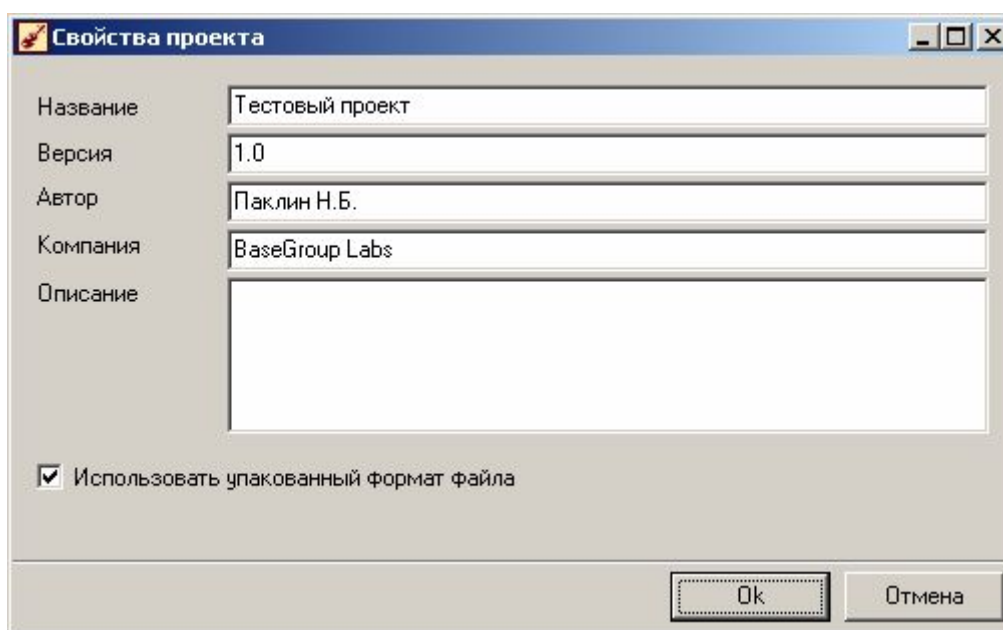
```
<?xml version="1.0" encoding="UTF-8"?>
<Document>
  <Version>
    <Comments>Deductor Studio Enterprise</Comments>
    <CompanyName>BaseGroup Labs</CompanyName>
    <FileDescription>Deductor Studio Enterprise</FileDescription>
    <FileVersion>5.2.0.50</FileVersion>
    <InternalName>Deductor Studio Enterprise</InternalName>
    <LegalCopyright>BaseGroup</LegalCopyright>
    <LegalTrademarks>BaseGroup</LegalTrademarks>
    <OriginalFilename>DStudio.exe</OriginalFilename>
    <ProductName>Deductor Studio Enterprise</ProductName>
```

<ProductVersion>5.2</ProductVersion>
</Version>

Замечание

*По умолчанию файл проекта **Deductor** при сохранении запаковывается, что позволяет уменьшить его размер, поэтому просмотреть запакованный файл в виде **xml** невозможно. Для этого нужно снять опцию **Использовать упакованный формат файла** в диалоговом окне **Свойства проекта** (меню **Файл ► Свойства проекта...**)*

Каждый проект имеет авторские сведения: Название, Версия, Автор, Компания, Описание. Они заполняются в диалоговом окне **Свойства проекта** (меню **Файл ► Свойства проекта...**).



Создать новый проект можно следующими способами:

- главное меню **Файл ► Создать**;
- кнопка **Создать новый проект** на панели инструментов;
- клавиша **Ctrl+N**.

Открытие существующего проекта:

- главное меню **Файл ► Открыть**;
- кнопка **Открыть проект** на панели инструментов;
- клавиша **Ctrl+O**.

Открыть проект можно еще одним способом – в главном меню **Файл ► История** найти имя проекта. Способ работает в том случае, если вы недавно открывали этот проект, и он сохранился в менеджере истории проектов.

В одной запущенной копии Deductor Studio можно открыть только один проект.

Для сохранения проекта под текущим именем нужно выбрать главное меню

Файл ► Сохранить, нажать кнопку  или комбинацию **Ctrl+S**.

Для сохранения текущего проекта под другим именем: главное меню **Файл ► Сохранить как...**

Мастера

В Deductor Studio вся работа ведется с использованием пяти мастеров:

- Мастер импорта;
- Мастер экспорта;
- Мастер обработки;
- Мастер визуализации;
- Мастер подключений.

С помощью мастеров импорта, экспорта и обработки формируется сценарий. Сценарий состоит из узлов. Мастер подключений предназначен для создания настроек подключений к различным источникам и приемникам данных. Мастер визуализации настраивает визуализаторы для конкретного узла.

Визуализатором называется любое представление набора данных в каком-либо виде: табличном, графическом, описательном. Примеры визуализаторов: таблица, дерево, гистограмма, диаграмма, OLAP-куб и т.д.

Практическая работа:

- 1 Создайте новый проект и сохраните его под именем **test.ded**. Не используйте упакованный формат файла.
- 2 Заполните свойства проекта.
- 3 Просмотрите файл проекта через любой текстовый редактор.
- 4 Сделайте видимой вкладку **Подключения**.
- 5 Поменяйте местами порядок вкладок **Сценарии** и **Подключения**.
- 6 Найдите в помощи раздел «Системные требования».

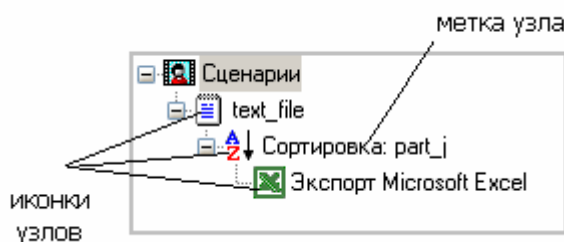
Вопросы для проверки:

- 1 Сколько вкладок на панели управления Deductor Studio?
- 2 Что такое проект в Deductor Studio?
- 3 Какое расширение имеет файл проекта?
- 4 Как создать новый проект?
- 5 Как сохранить текущий проект под другим именем?
- 6 Как отредактировать свойства проекта?
- 7 Сколько проектов можно одновременно открыть в Deductor Studio?
- 8 Сколько мастеров имеется в Deductor Studio?

Занятие 3. Сценарии

Понятие сценария и узла обработки

В Deductor Studio для аналитика основополагающим понятием является сценарий. Сценарий представляет собой последовательность операций с данными, представленную в виде иерархического дерева. В дереве каждая операция образует узел, заголовок которого содержит: имя источника данных, наименование применяемого метода обработки, используемые при этом поля и т.д. Кроме этого, слева от наименования узла стоит значок, соответствующий типу операции.



Если узел имеет подчиненные узлы, то слева от его названия будет расположен значок «+», щелчок по которому позволит развернуть узел, т.е. сделать видимыми все его подчиненные узлы, при этом значок «+» поменяется на «-». Щелчок по значку «-», наоборот, сворачивает все подчиненные узлы.

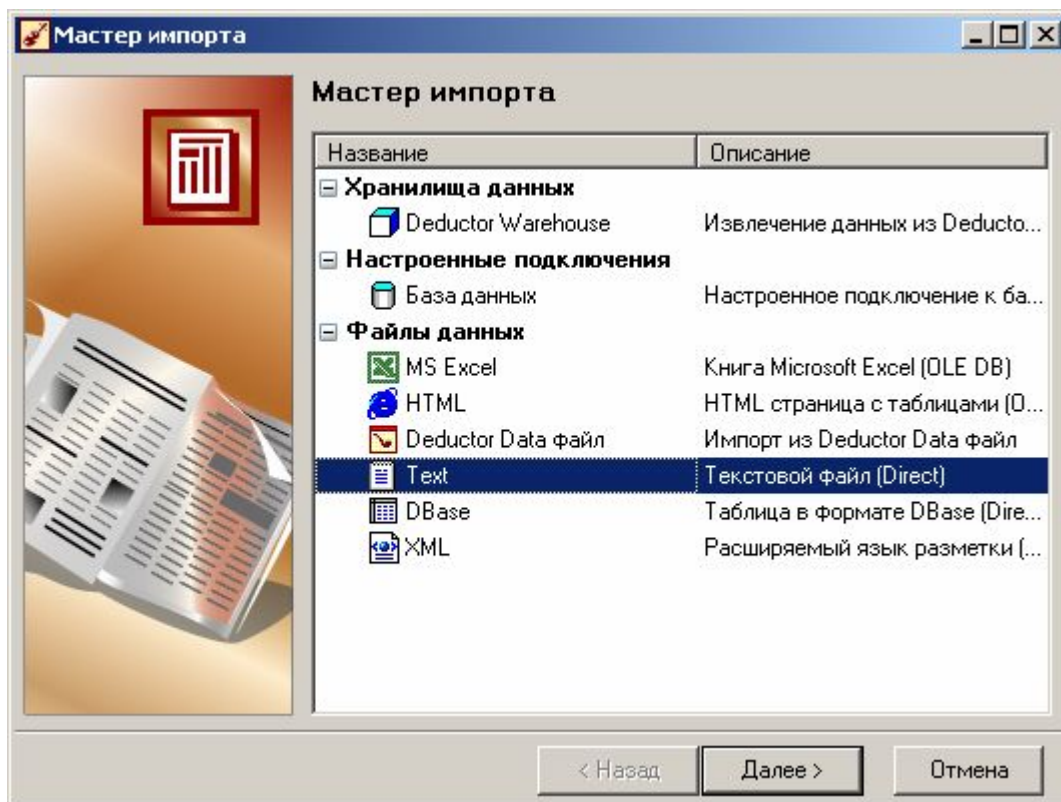
С помощью клавиш **Ctrl+↑** и **Ctrl+↓** можно перемещать узлы по дереву вверх-вниз в пределах подчинения родительскому узлу.

Сценарий состоит из **ветвей**. **Deductor** не имеет собственных средств для ввода данных, поэтому сценарий **всегда** начинается с узла импорта из какого-либо источника. Любой вновь создаваемый узел импорта будет находиться на верхнем уровне (подчиненным главному узлу Сценарии).

Создание нового узла импорта осуществляется с помощью **мастера импорта**. Вызвать мастер можно следующими способами:

- кнопка  на панели инструментов закладки **Сценарии**;
- клавиша **F6**;
- контекстное меню **Мастер импорта...**

При вызове мастера импорта откроется окно первого шага мастера.



В нем все источники данных сгруппированы по следующим четырем категориям:

- хранилища данных;
- настроенные подключения;
- файлы данных;
- бизнес-подключения.


.Некоторые категории могут отсутствовать в списке. Причинами этого может быть следующее:

- Версия Deductor. Например, категории **Настроенные подключения** и **Бизнес-подключения** отсутствуют в версии Academic.
- В дереве подключений (вкладка **Подключения**) не зарегистрировано ни одного объекта из данной категории. Например, если не настроено ни одного подключения к хранилищу данных, то категория **Хранилища данных** будет отсутствовать.
- Отключена «видимость» объекта или категории объекта (подробнее об этом см. в разделе **Настройка конфигурации Deductor Studio** в Занятии 9).

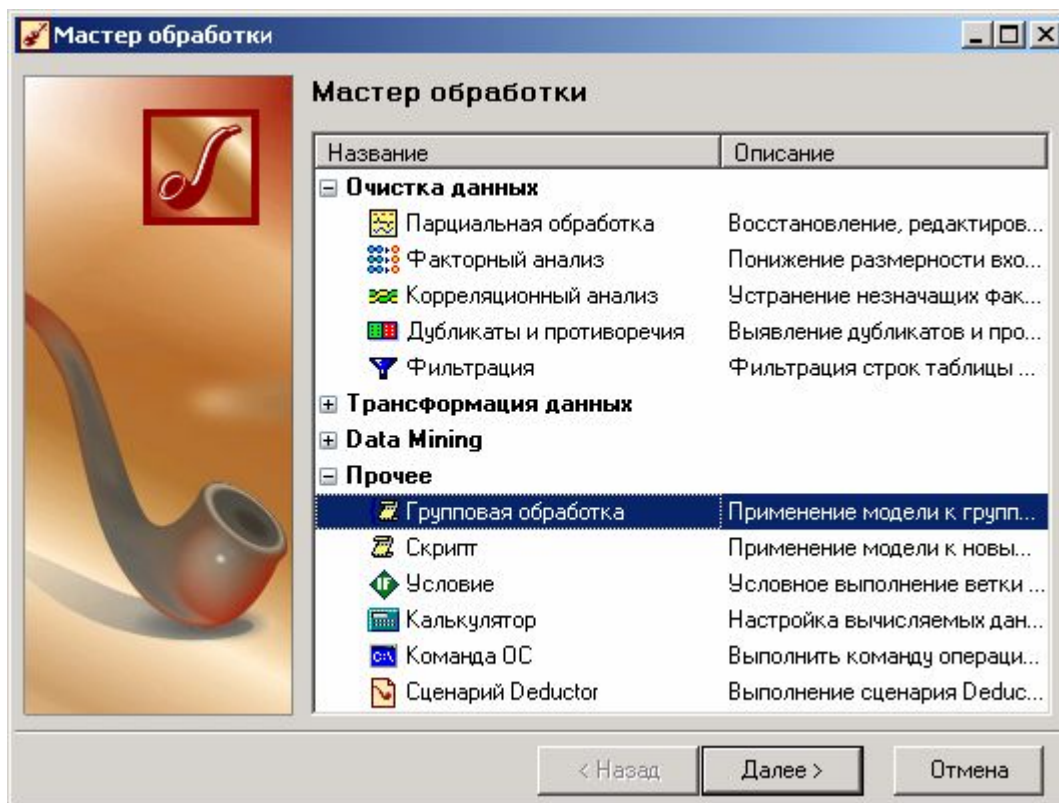
Дальнейшие шаги мастера импорта будут зависеть от того, какой объект дерева категорий был выбран аналитиком.

К любому узлу импорта можно добавить узел **обработки** или узел **экспорта**, предварительно выделив узел импорта мышью. Новый узел будет добавлен как подчиненный к узлу импорта.

Создание нового узла обработки осуществляется с помощью **мастера обработки**. Вызвать мастер можно следующими способами:

- кнопка  на панели инструментов закладки **Сценарии**;
- клавиша **F7**;
- контекстное меню **Мастер обработки...**

При вызове мастера обработки откроется окно первого шага мастера.




В нем все обработчики сгруппированы по следующим четырем категориям:

- Очистка данных;
- Трансформация данных;
- Data Mining;
- Прочее.

Некоторые узлы могут отсутствовать в списке. Причины этого следующие:

- версия Deductor;
- отключена «видимость» объекта (или целой категории) объекта;
- узел «устарел» и в текущей версии Deductor его создание невозможно (допускается только его чтение и настройка).

Создание нового узла экспорта осуществляется с помощью **мастера экспорта**. Вызвать мастер можно следующими способами:

- кнопка  на панели инструментов закладки **Сценарии**;
- клавиша **F8**;
- контекстное меню **Мастер экспорта...**

В нем все приемники данных сгруппированы по следующим 5 категориям:

- хранилища данных;
- базы данных;
- файлы;
- Web-серверы;

- прочее.

Причины отсутствия некоторых объектов или категорий мастера экспорта аналогичны тем, что перечислены при описании мастера импорта.



После узла экспорта невозможно добавить ни один узел.

Базовые операции над узлами сценария

Кроме команд вызова мастеров, к каждому узлу применимы базовые операции. Операции над узлами и ветками сценария можно выполнять следующими способами:

- кнопки панели инструментов на закладке **Сценарии**;
- контекстное меню;
- мышь.


Список доступных операций.

- 1 **Открытие узла** – узел запускается на выполнение, причем выполняются все родительские узлы, а справа открываются визуализаторы, настроенные для данного узла. В интерактивном режиме для каждого узла должен быть настроен хотя бы один визуализатор, например, Таблица или Сведения. Операция вызывается:
 - двойной щелчок мышью на узле;
 - клавиши **Ctrl+Enter**;
 - контекстное меню Открыть.
- 2 **Настройка узла** – вызывается мастер импорта, мастер обработки или мастер экспорта, в зависимости от типа узла, для изменения параметров обработки, производимой в узле. Операция вызывается:
 - кнопка ;
 - клавиши **Alt+Enter**;
 - контекстное меню Настроить....
- 3 **Активация/деактивация узла** – узел может быть либо активным, либо неактивным. Если узел неактивный, то, сделав его активным, выполнится сценарий для этого узла, но визуализаторы отображены не будут. Делая узел неактивным, закрываются все визуализаторы для него и для всех подчиненных узлов, а сам узел и подчиненные узлы превращаются в неактивные. Эта операция может быть использована для освобождения памяти. Операция активации/деактивации вызывается:
 - клавиши **Shift+Enter**;
 - контекстное меню Активный...
- 4 **Перечитать данные узла** – все узлы до корневого включительно будут закрыты, а затем выполнена ветка сценария от корневого до текущего узла. Операция вызывается:
 - контекстное меню Перечитать данные...
- 5 **Вырезать узел** – удаляет текущий узел из сценария обработки. Все его потомки при этом перемещаются на один уровень вверх и начинают подчиняться родителю удаленного узла. Операция вызывается:
 - кнопка ;
 - контекстное меню Вырезать узел.
- 6 **Вставить узел** – вставляет перед текущим узлом сценария новый узел и вызывает для него мастер обработки. Вставить узел перед узлом импорта данных нельзя. Операция вызывается:

- кнопка ;
- контекстное меню **Вставить узел**.

После вставки нового узла или удаления существующего узлы-потомки могут стать неработоспособными, в зависимости от обработки, выполняемой новым узлом.

7 Копировать ветвь – копирует ветвь сценария, начиная с текущего узла и включая все его потомки. Операция вызывается:

- кнопка ;
- контекстное меню **Копировать ветвь**;
- при помощи механизма drag & drop – выделив узел, и, удерживая нажатой клавишу **Ctrl**, указать курсором мыши на новый узел, который должен стать родителем старого. При этом переносимая ветка целиком скопируется в новое место.

8 Удалить ветвь – удаляет узел сценария и все его подузлы. Удаленная ветвь восстановлению не подлежит, поэтому к данной операции необходимо подходить с осторожностью. Операция вызывается:


- кнопка ;
- клавиши **Ctrl+Del**;
- контекстное меню **Удалить ветвь**.

9 Перенос ветви – переносит ветку сценария к новому узлу. Операция производится аналогично копированию ветви с помощью drag & drop без удерживания клавиши **Ctrl**.

10 Переименовать – позволяет изменить метку текущего узла. Операция вызывается:

- клавиша **F2**;
- контекстное меню **Переименовать...**

11 Сведения – открывает диалоговое окно **Сведения** для текущего узла. В нем редактируется имя, метка и описание к узлу. Операция вызывается:

- контекстное меню **Сведения...**;
- открыв скрытую панель узла с помощью кнопки  и нажать там одну из кнопок: **Имя**, **Метка** или **Описание**.

Имя узла может быть задано только латинскими символами, тогда как метка – любыми. Кроме того, имя узла должно быть уникально в пределах одного сценария. Как правило, необходимости в переименовании имен узлов не возникает.

12 Статус пакетной обработки – устанавливает статус пакетной обработки для узла.

13 Добавить в Избранное – текущий узел добавляется в список избранных узлов.

14 Сохранение ветви – вызывается стандартный диалог **Сохранение**, в котором можно указать путь и имя файла для сохранения ветви сценария, начинающейся с текущего узла. Операция вызывается:

- контекстное меню **Сохранить ветвь**.

15 Загрузка ветви – вызывает стандартный диалог **Открытие файла**, в котором можно указать путь и имя файла, хранящего ветвь сценария. Загруженная ветвь сценария станет потомком текущего узла. Ветвь, начинающаяся с узла импорта данных, будет добавлена в проект как новая корневая ветвь. Операция вызывается:

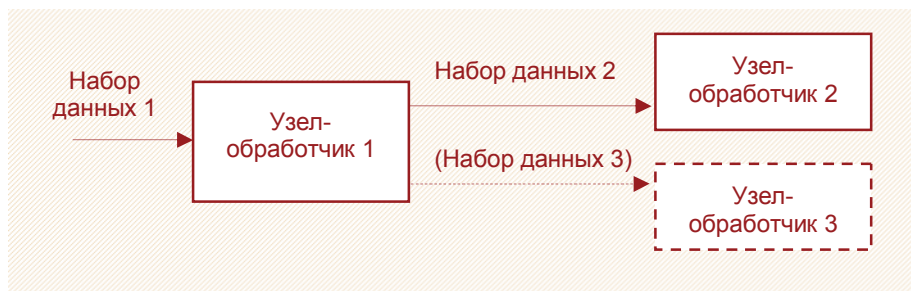
- контекстное меню **Загрузить ветвь**.

По умолчанию ветвь сценария имеет расширение ***.deb**.

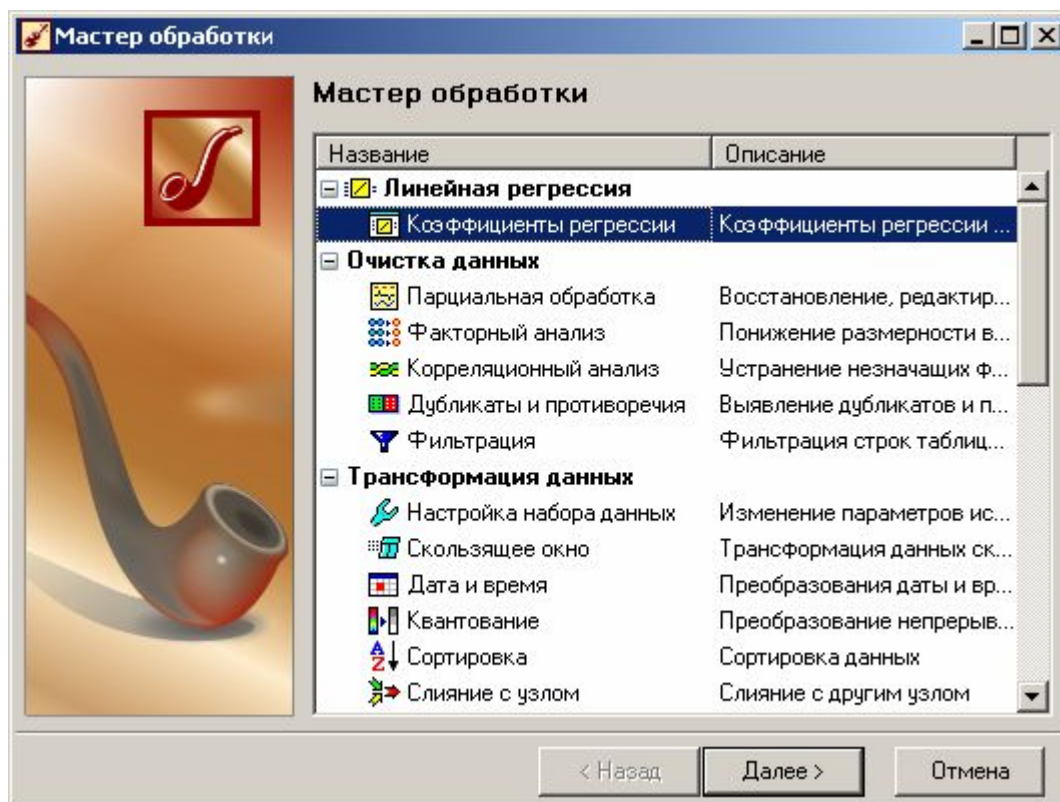
Взаимодействие узлов друг с другом

В Deductor взаимодействие узлов друг с другом спроектировано на уровне программного ядра, поэтому принцип взаимодействия един и не зависит от типа узла.

Каждый узел можно представить «черным ящиком», на вход которого подается структурированный набор данных с полями, а на выходе доступен один или несколько обработанных узлов наборов данных. Обработка может вестись любая – от простой сортировки до моделирования. Выходной набор, в свою очередь, можно снова подать на вход узла. Так конструируется сценарий.



Но иногда на выходе узла может присутствовать не один набор, а несколько (на рисунке такой дополнительный набор данных обозначен пунктирной стрелкой). Например, в результате работы узла **Линейная регрессия** образуются два набора данных: один – таблица рассчитанных результатов, а другой – коэффициенты регрессии. Эти коэффициенты можно просмотреть в визуализаторе под таким же названием, но иногда нужно использовать коэффициенты в сценарии для дальнейшей обработки. Поэтому при добавлении любого узла появляется возможность «переключиться» на другой набор данных, если он присутствует в предыдущем узле. Вот как это выглядит в мастере обработки.




В Deductor Studio 5.2 узлами, которые выдают на выходе более одного набора данных, являются: **Линейная регрессия, Логистическая регрессия, Ассоциативные правила, Корреляционный анализ.**

Импорт из текстовых файлов с разделителями

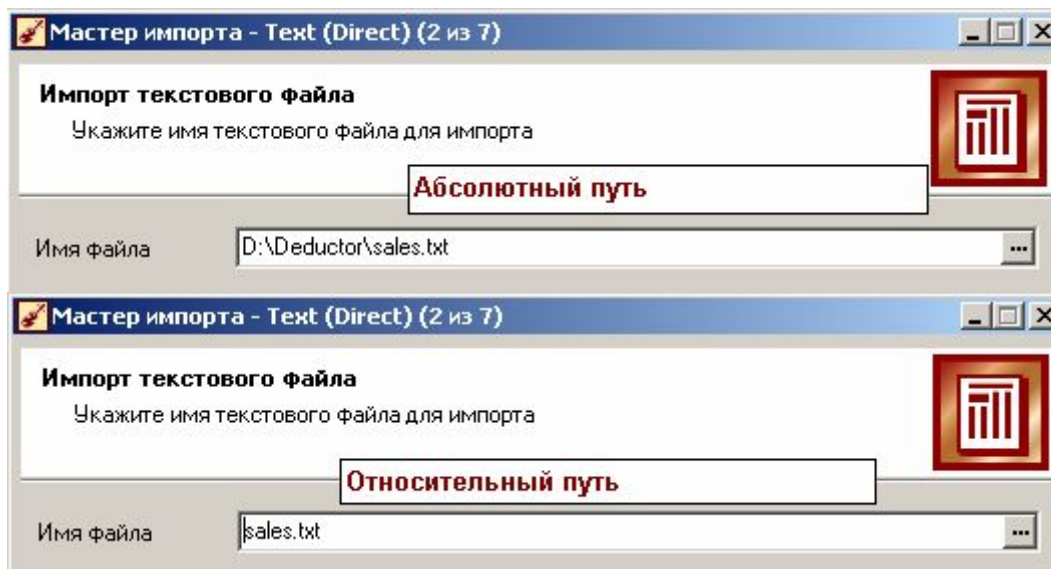
Структурированный текстовый файл с разделителями – один из самых распространенных форматов хранения данных. Такой файл представляет собой обычный текстовый файл, столбцы данных в котором разделены однотипными символами-разделителями, например символами табуляции, пробела, точки с запятой и т.д.

Процесс импорта данных из текстового с разделителями файла в мастере импорта (категория **Текстовый файл (Direct)**) содержит следующие шаги:

- указание имени файла;
- настройка параметров импорта;
- настройка импортируемых полей;
- запуск процесса импорта;
- выбор способа визуализации;
- задание сведений об узле.

На шаге **Указание имени файла**, нажав кнопку , необходимо выбрать имя текстового файла (расширения *.txt, *.csv), из которого следует выполнить импорт данных. После этого в поле «Имя файла» окна Мастера импорта появится имя выбранного файла и путь. Допускается вручную ввести путь к файлу в строке поля Имя файла.

Имеется возможность использовать как абсолютные, так и относительные пути для файлов. Они указываются относительно текущей директории Deductor. При открытии Deductor текущей директорией является директория файла проекта. Поэтому, если файл проекта и текстовые файлы располагаются в одной папке, то использование относительных путей в Мастере импорта позволит не перенастраивать узлы импорта при изменении расположения папки на жестком диске.



Здесь также доступны настройки:

- **Начать импорт со строки** – номер строки, начиная с которой будет делаться импорт данных из файла.

- флаг **Первая строка является заголовком** – установка флажка означает, что узел будет импортировать данные с учетом того, что все записи первой строки являются заголовками столбцов.
- **Кодировка** – ANSI (Windows) или ANCI (MS DOS).

На шаге **Настройка параметров импорта** нужно настроить параметры импорта данных из текстового файла, так как существует несколько форматов структурированных текстовых файлов. Доступные опции:

- переключатель **Формат исходных данных**, который определяет символ-разделитель в файле (например: символ табуляции, пробел, запятая). Разделитель чаще всего присутствует. Если же нет, то нужно выбрать переключатель **Фиксированной ширины (поля имеют заданную ширину)**, а позже установить ширину каждого поля.
- **Ограничитель строк** – при задании данного параметра необходимо указать, какой именно ограничитель строкового значения нужно использовать при импорте данных из текстового файла. Обычно таким ограничителем является символ двойной кавычки " .
- **Разделитель дробной и целой части числа** – при задании данного параметра необходимо указать символ, разделяющий дробную и целую части в числовых значениях, содержащихся в файле.
- **Разделитель компонентов даты** – указывается символ, разделяющий компоненты даты в соответствующих значениях, содержащихся в файле.
- **Разделитель компонентов времени** – указывается символ, разделяющий компоненты времени в соответствующих значениях, содержащихся в файле.
- **Форматы Даты/Времени** – указываются форматы даты/времени, используемые в импортируемом файле.
- **Представление значений** – опция для полей логического типа, которое может принимать одно из трех значений – истина (true), ложь (false) и пустое значение (null). Определяет регламент записи в эти значения. Так, при настройках по умолчанию для любого логического поля значение Да будет восприниматься как истина, Нет – как ложь.

В качестве разделителей, представлений значений и форматов по умолчанию *всегда предлагаются* системные настройки операционной системы. Поэтому при импорте необходимо обращать внимание на их соответствие формату в импортируемом текстовом файле.

Следующее окно мастера зависит от установленного переключателя в флажке **Формат исходных данных**. Если был выбран формат **С разделителями**, то появится вкладка, на которой нужно явно указать символ-разделитель (по умолчанию – табуляция). Здесь же находится флаг **Считать последовательные разделители одним** – в случае последовательно идущих символов-разделителей они будут восприниматься за один. Такое бывает, например, когда символом-разделителем выступают несколько пробелов.

Предпросмотр текстового файла в виде таблицы внизу (загружаются только первые 10 строк) позволяет убедиться в корректности выбора настроек импорта даже не запуская его.

Мастер импорта - Текст (4 из 9)

Параметры импорта файла с разделителями

Укажите символ-разделитель столбцов и другие вспомогательные параметры импорта

Символом-разделителем является:

☒ Символ табуляции ☐ Пробел ☐ Точка

☐ Точка с запятой ☐ Запятая ☐ Другой

☐ Считать последовательные разделители одним

Банк	Активы			Структура п	
	Вложения в векселя, %	Кредиты частным лицам, %	Кредиты предприятиям и организациям, %	Собственный капитал, %	ривлеченный МБК, %
Сбербанк	0	23	56	10	3
ВТБ	2	1	46	17	31
Газпромбанк	1	3	40	12	14

< Назад Далее > Отмена

Если был выбран флаг формат **Фиксированной ширины**, то появится вкладка, на которой нужно задать границы каждого поля. Создание, как и удаление маркера границы производится одним щелчком мыши. Двигая маркеры границ столбцов, можно изменять их, если они расставлены неправильно. Данные, распределенные по столбцам, показываются в области предварительного просмотра.

Мастер импорта - Текст (5 из 9)

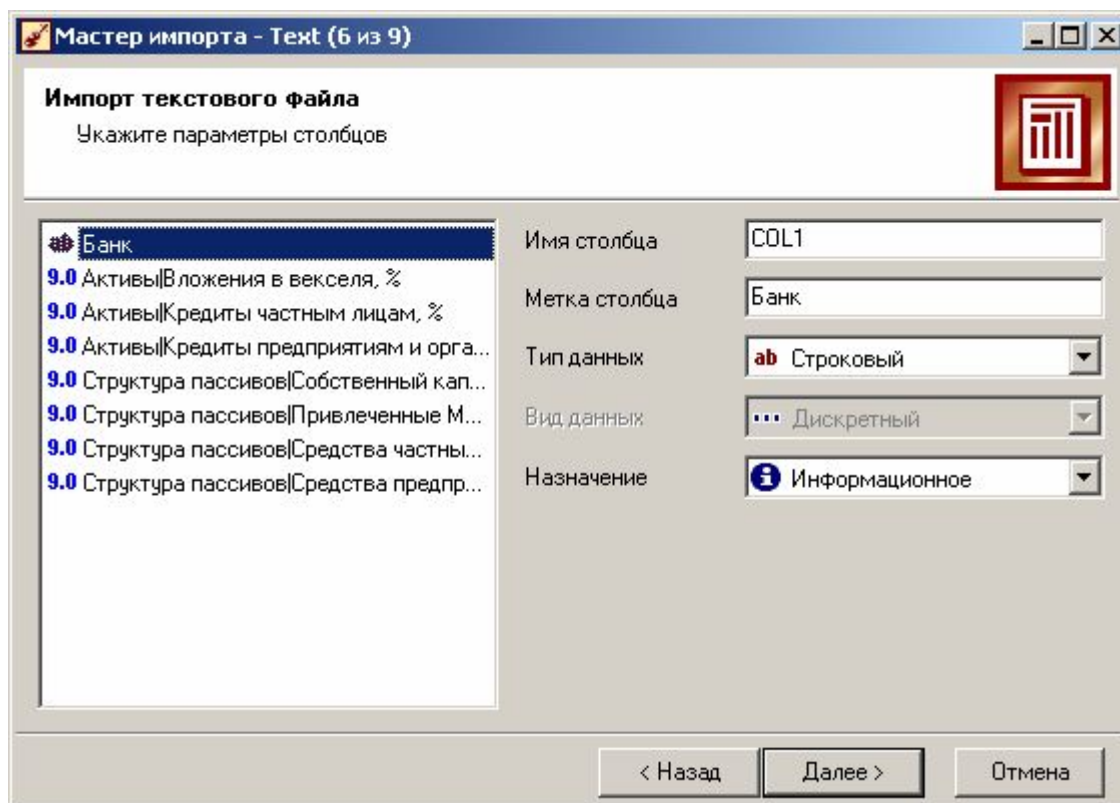
Параметры импорта файла со столбцами фиксированной ширины

Линии со стрелками обозначают границы столбца;
ВСТАВИТЬ/УДАЛИТЬ разделитель - двойной щелчок

Банк	Активы	Вложения в векселя, %	Ак
Сбербанк		0	23
ВТБ		2	1
Газпромбанк		1	3
Альфа-банк		0	6
Банк Москвы		0	10
Уралсиб		1	15
Росбанк		1	29
Россельхозбанк		2	13
ММБ		0	10
Райффайзенбанк		0	13
Русский стандарт		0	76

< Назад Далее > Отмена



На шаге **Настройка параметров столбцов** нужно настроить следующие параметры столбцов импортируемых данных, указав соответствующие значения в полях.



Имя столбца – указывается имя, которое будет служить идентификатором столбца в последующих узлах. По умолчанию предлагается заголовок столбца из текстового файла, если на предыдущем шаге был установлен флажок **Первая строка является заголовком**. Иначе будут предложены имена типа COL1, COL2 и т.д. Можно ввести любые имена, которые семантически отражают содержимое столбца, однако допускаются только латинские символы, и имя столбца должно быть уникальным в пределах всех столбцов импортируемого файла.

Метка столбца – название, под которым данный столбец будет виден в визуализаторах. Допускаются любые символы, уникальность имен не обязательна.

Тип данных – указывается тип данных, содержащихся в столбце. Тип выбирается из списка, открываемого щелчком по кнопке в правой части поля:

Тип	Описание
 логический	данные в поле могут принимать только два значения – 0 или 1
 дата/время	поле содержит данные типа дата/время
9.0 вещественный	числа с плавающей точкой
12 целый	целые числа
ab строковый	строки символов

Узел импорта всегда пытается автоматически распознать тип данных *по первой строке файла* (если имеются заголовки, то *по второй строке*). Такой алгоритм срабатывает не всегда. К примеру, пусть в файле есть столбец Число иждивенцев, и в нем данные идут в следующем порядке:

Иждивенцы
2
1
нет
2
более 2

Для данного поля автоматически определится тип – *вещественный*, но в реальности он *строковый*.










Вид данных – характер данных, содержащихся в столбце:

Вид	Описание
<i>непрерывный</i>	значения в столбце могут принимать любое значение в рамках своего типа
<i>дискретный</i>	данные в столбце могут принимать ограниченное число значений

Непрерывными могут быть только числовые данные. Дискретный характер носят, как правило, строковые данные, но не всегда. Дискретными могут быть назначены в зависимости от контекста решаемой задачи данные целого типа, реже – вещественного. Вид данных столбца влияет на:

- алгоритм расчета статистики по столбцу;
- работу аналитических алгоритмов.

Назначение – определяет порядок использования поля набора данных, полученного в результате импорта столбца (поля), при дальнейшей обработке импортированных данных:

Назначение	Описание
 <i>первичный ключ</i>	поле будет использоваться в качестве первичного ключа
 <i>входное</i>	поле набора данных, построенное на основе столбца, будет являться входным полем обработчика (нейронной сети, дерева решений и т.д.)
 <i>выходное</i>	поле набора данных, построенное на основе столбца, будет являться выходным полем обработчика (например, целевым полем для обучения нейронной сети).
 <i>информационное</i>	поле содержит вспомогательную информацию, которую часто полезно отображать, но не следует использовать при обработке
 <i>измерение</i>	поле будет использоваться в качестве измерения в многомерной визуализации
 <i>атрибут</i>	поле содержит описание свойств или параметров некоторого объекта
 <i>факт</i>	значения поля будут использованы в качестве фактов в многомерной визуализации
 <i>транзакция</i>	транзакция – поле, содержащее идентификатор событий, происходящих совместно (одновременно); например, номер чека, по которому приобретены товары
 <i>элемент</i>	поле, содержащее элемент транзакции (событие).

Изменить назначение группы столбцов одной операцией можно следующим образом:

- удерживая клавишу **Shift**, выделить мышкой или клавишами **Ctrl+↓**, **Ctrl+↑** первый и последний столбцы группы столбцов и изменить их назначение;
- удерживая клавишу **Ctrl**, выделить мышкой только нужные столбцы и изменить их назначение.

Замечание

*Установка назначения столбца набора данных при импорте не является обязательным действием (по умолчанию при импорте установлено назначение «информационное»). Однако это может снизить объем рутинных действий при последующем конструировании сценария. Например, при построении моделей (группа узлов обработки **Data Mining**) по умолчанию выходным полем, как правило, предлагается последнее поле, и, если это не так, придется каждый раз переопределять назначения полей в каждом новом узле.*

На шаге **Запуск процесса импорта** стартует сам процесс импорта данных с ранее настроенными параметрами. Ход процесса импорта отображается с помощью индикатора. Если процесс импорта остановился, это сигнализирует о возможных ошибках при чтении данных. В этом случае появляется окно с сообщением об ошибке.

В случае возникновения ошибок несоответствия типов процесс импорта будет продолжен, но после его окончания будет отображен журнал регистрации ошибок с информацией о месте и причине их появления:

Deductor Studio Enterprise 5.2.0.50

29.07.2008 13:19:24 TBGTextFile: При разборе строки 2 возникла ошибка:
В колонке "Группа.Код" значение "33" не удалось преобразовать к дате/времени.
29.07.2008 13:19:24 TBGTextFile: При разборе строки 3 возникла ошибка:
В колонке "Группа.Код" значение "48" не удалось преобразовать к дате/времени.
29.07.2008 13:19:24 TBGTextFile: При разборе строки 4 возникла ошибка:
В колонке "Группа.Код" значение "50" не удалось преобразовать к дате/времени.
29.07.2008 13:19:24 TBGTextFile: При разборе строки 5 возникла ошибка:
В колонке "Группа.Код" значение "108" не удалось преобразовать к дате/времени.

Для управления процессом импорта предусмотрены следующие кнопки:

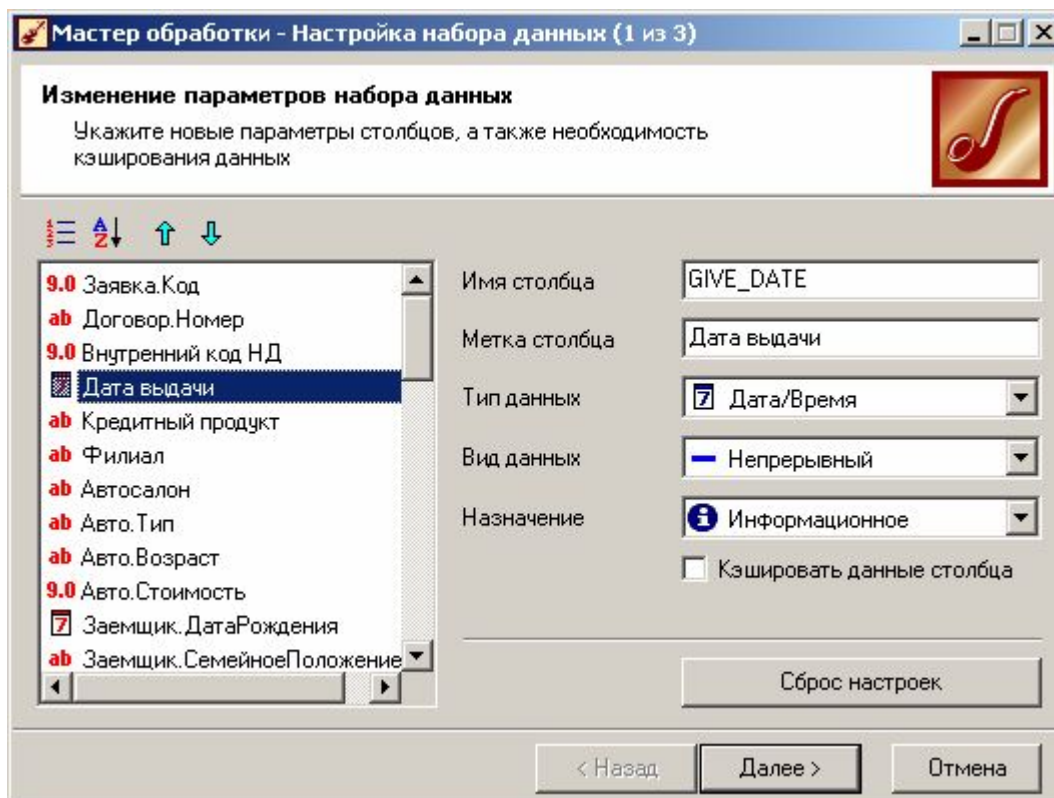
- Пуск – запускает процесс в первый раз или возобновляет после паузы.
- Пауза – временно приостанавливает импорт.
- Стоп – останавливает процесс без возможности его продолжения.

На оставшихся двух шагах мастера импорта будет предложено выбрать визуализатор набора данных (по умолчанию предлагается **Таблица**) и задать сведения об узле.

Узел Настройка набора данных

Обработчик **Настройка набора данных** позволяет:

- изменить имя, метку, тип, вид и назначение полей текущего набора данных;
- изменить порядок следования столбцов в наборе данных;
- скрыть столбцы набора данных;
- задать опцию кэширования выходного набора.





Изменение имени или метки поля удобно в тех случаях, когда имена столбцов могут измениться в источнике данных или при перенастройке узлов верхних уровней. В этом случае в узле **Настройка набора данных** имя исходного столбца заменяется другим, на которое и настраиваются все дочерние узлы. После такой операции изменение имен полей на верхних уровнях не потребует перенастройки всех дочерних узлов в дереве сценариев.

Тип, вид и назначение можно изменить у нескольких столбцов одной операцией. Для этого достаточно их выделить, удерживая нажатой клавишу **Ctrl** или **Shift**.

Если параметры столбца были изменены, цвет иконки столбца меняется на **красный**. Для установки первоначальных параметров столбцов необходимо выделить столбец или список столбцов и нажать на кнопку **Сброс параметров**.

Чтобы скрыть столбец из набора данных, нужно задать ему назначение

✗ Неиспользуемое.

Изменить порядок следования столбцов в наборе данных можно при помощи клавиш  .

Кэширование – это загрузка часто используемой информации в оперативную память для быстрого доступа к ней, минуя многократные считывания с жесткого диска. Кэширование может заметно повысить скорость работы сценария в ряде случаев (использование кэширования не входит в базовые навыки работы с Deductor).

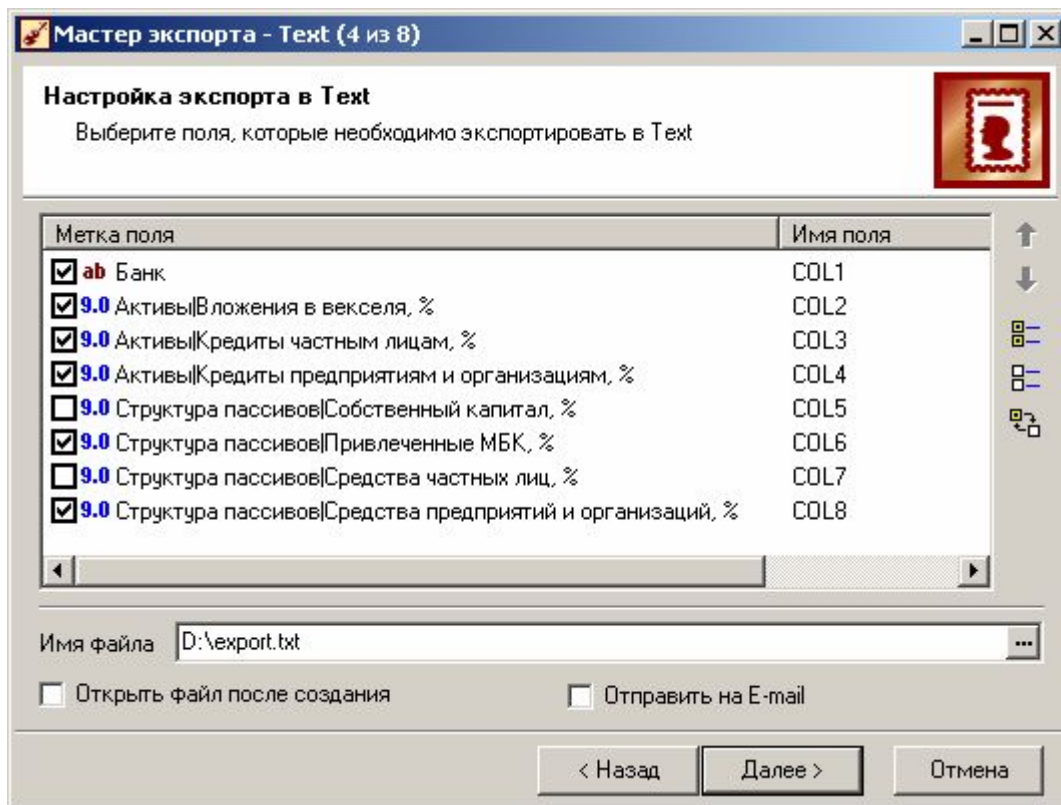
Экспорт в текстовый файл

Выполняется при помощи мастера экспорта. В нем процесс экспорта данных в текстовый файл с разделителями (категория **Файлы**) содержит следующие шаги:

- настройка форматов экспорта;
- указание символа-разделителя столбцов;
- выбор экспортируемых полей;

- запуск процесса экспорта;
- выбор способа визуализации;
- задание сведений об узле.

На шаге **Настройка параметров экспорта** задаются параметры экспорта данных из текстового файла аналогично тем, что задавались в мастере импорта. Экспортироваться будут не все поля, а только те, у которых поднят флажок на шаге **Выбор экспортируемых полей**:



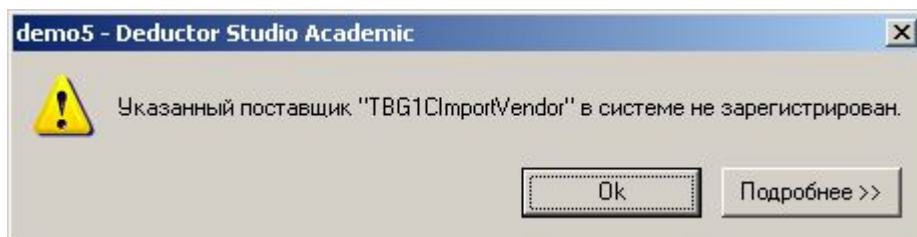
Здесь же задается имя файла экспорта. По умолчанию предлагается имя файла **export.txt**. Как и в случае с импортом, допускается использовать относительные пути.


Флажок **Открыть после создания** откроет текстовый файл программой-просмотрщиком. Установка **флага Отправить на E-mail** позволит отправить файл на почтовый адрес в виде вложенного в письмо файла (доступно только в версии **Enterprise**).

Вкладка запуска процесса экспорта аналогична той, что присутствовала в мастере импорта. Если файл с таким именем уже существует, возникнет окно с подтверждением о перезаписывании этого файла. На шаге выбора способа визуализации будет доступен только один визуализатор **Прочее**. Задание сведений об узле завершит работу мастера экспорта.

Узел «!»

При открытии сценария некоторые узлы могут иметь значок **!**, а при попытке выполнить ветвь узла появится, например, следующее сообщение:



Значок  говорит о том, что выполнить данный узел невозможно. Причинами этого могут быть:

- Узел не поддерживается в текущей поставке Deductor. Например, узлы импорта из 1С не выполняются в Deductor Academic.
- Сценарий создан в более поздней версии (сборке) Deductor, а открыт в более ранней, и функционал такого узла еще не существовал в ранних версиях. Номер сборки можно проверить, открыв меню ? ► **О программе**.

Практическая работа:

- 1 Создайте новый проект и сохраните его под именем **test2.ded**.
- 2 Создайте и сохраните в любом текстовом редакторе файл следующего вида:

a,1,4.5,b,c,26/04/2007,d

a1,0,5,b1,c1,,d1
- 3 Импортируйте его в Deductor, корректно настроив параметры импорта. Используйте относительный путь для файла. Метку узла переименуйте в Пример импорта файла. В комментарии к узлу впишите: Текстовый файл с разделителями-запятыми.
- 4 Добавьте к узлу узел Настройка набора данных и задайте следующие метки к столбцам: Поле1, Поле2, Поле3 и т.д.
- 5 Экспортируйте набор данных в текстовый файл с настройками, предлагаемыми по умолчанию.
- 6 Импортируйте только что экспортированный файл в Deductor.
- 7 Присоедините к новому узлу импорта (путем копирования) предыдущую ветвь, начиная с узла **Настройка набора данных**.
- 8 Между экспортом и настройкой набора данных вставьте еще один узел настройки, в котором измените тип столбца Поле2 на логический.
- 9 Удалите только что вставленный узел.
- 10 Сохраните проект.

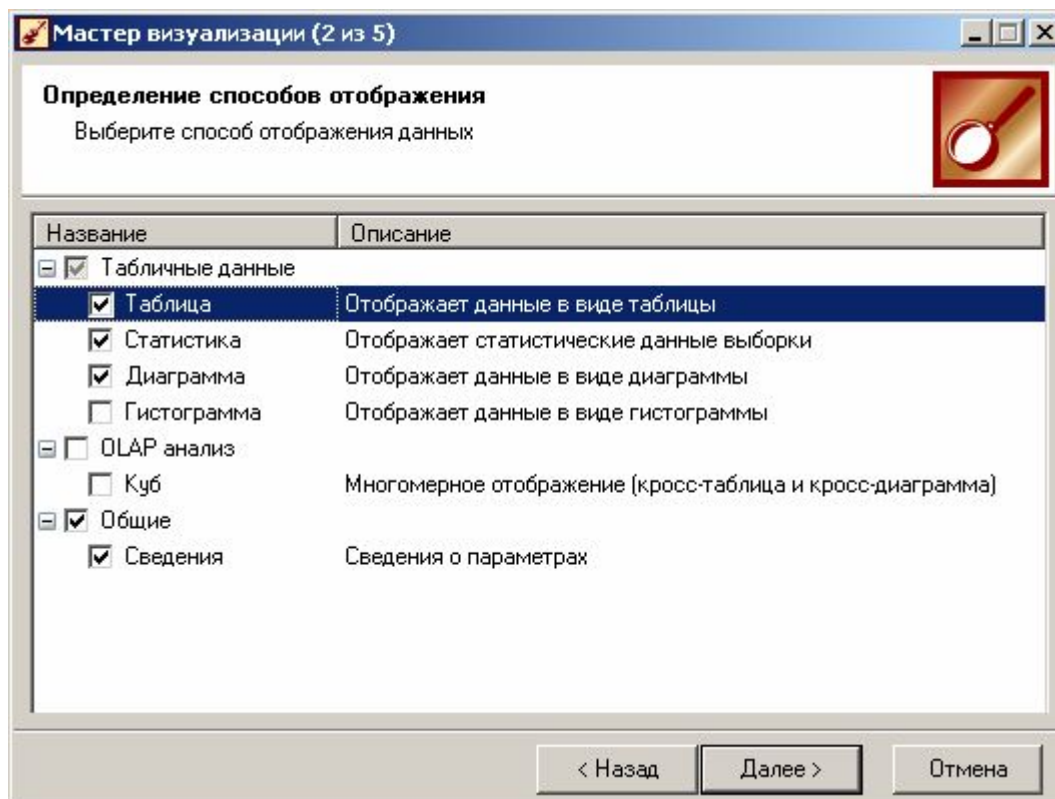
Вопросы для проверки:

- 1 Сколько вкладок на панели управления Deductor Studio?
- 2 Что такое «проект» в Deductor Studio?
- 3 Какое расширение имеет файл проекта?
- 4 Как создать новый проект?
- 5 Как сохранить текущий проект под другим именем?
- 6 Сколько проектов можно одновременно открыть в Deductor Studio?
- 7 Сколько мастеров имеется в Deductor Studio?


- 8 Вы импортировали текстовый файл, создав узел импорта. После чего обнаружили, что неправильно задали параметры импорта. Как легче всего исправить ошибку?
- 9 Как скопировать ветвь сценария при помощи механизма drag & drop?
- 10 Какие шаги мастера импорта нужно пройти для импорта текстового файла?
- 11 Что позволяет сделать обработчик **Настройка набора данных**?
- 12 Как происходит автоматическое определение типа столбца при импорте?
- 13 Что означает пиктограмма «!» напротив узла сценария?

Занятие 4. Базовые визуализаторы

К каждому узлу сценария, который содержит структурированный набор данных, всегда предлагается несколько визуализаторов. **Мастер визуализации** в интерактивном пошаговом режиме позволяет выбрать и настроить наиболее удобный способ представления данных. В зависимости от выбранного способа будут настраиваться различные параметры, а Мастер, соответственно, будет содержать различное число шагов. Первый шаг мастера визуализации будет одинаков для всех видов, поскольку на нем и производится выбор визуализатора.



Вызов мастера визуализации:

- кнопка  на панели инструментов закладки **Сценарии**;
- клавиша **F5**;
- контекстное меню **Мастер визуализации...**

Мастер визуализации запускается для выделенного узла сценария. Кроме того, этот мастер всегда является продолжением мастера обработки, т.е. активизируется при создании (настройке) любого узла.

Желаемые способы отображения следует пометить флажками. Одновременно может быть выбрано несколько визуализаторов, при этом каждый из них будет открыт в отдельном окне.

Замечание

*Если на первом шаге мастера визуализации одновременно выбрано несколько способов отображения данных, то все соответствующие шаги будут последовательно включены в общую процедуру настройки. Например, если выбраны **Таблица** и **Диаграмма**, то в мастер визуализации будут последовательно включены отдельные шаги для настройки таблицы и диаграммы.*

Базовыми визуализаторами в Deductor являются следующие:

- Таблица;
- Статистика;
- Сведения.

Визуализатор *Таблица*



Дата кредитования	Сумма кредита	Срок кредита	Цель кредитования	Частная собственность
01.01.2003	7000	6	Иное	<input type="checkbox"/>
01.01.2003	7500	6	Иное	<input checked="" type="checkbox"/>
01.01.2003	14500	12	Покупка товара	<input checked="" type="checkbox"/>
01.01.2003	15000	6	Покупка товара	<input type="checkbox"/>
01.01.2003	32000	12	Иное	<input checked="" type="checkbox"/>
01.01.2003	11500	6	Турпоездки, развлечения и т.п.	<input type="checkbox"/>
01.01.2003	5000	6	Покупка и ремонт недвижимости	<input checked="" type="checkbox"/>
01.01.2003	61500	30	Покупка товара	<input checked="" type="checkbox"/>
01.01.2003	13500	12	Оплата услуг (мед., юрид. и т.п.)	<input type="checkbox"/>
01.01.2003	25000	18	Покупка товара	<input checked="" type="checkbox"/>
01.01.2003	25500	24	Покупка товара	<input checked="" type="checkbox"/>
01.01.2003	9500	6	Покупка товара	<input checked="" type="checkbox"/>
01.01.2003	53000	24	Иное	<input checked="" type="checkbox"/>
02.01.2003	27500	18	Покупка товара	<input checked="" type="checkbox"/>
02.01.2003	4000	6	Оплата услуг (мед., юрид. и т.п.)	<input type="checkbox"/>
02.01.2003	40500	24	Покупка и ремонт недвижимости	<input checked="" type="checkbox"/>
02.01.2003	51500	36	Покупка и ремонт недвижимости	<input type="checkbox"/>
02.01.2003	7000	6	Оплата услуг (мед., юрид. и т.п.)	<input type="checkbox"/>
02.01.2003	8500	6	Турпоездки, развлечения и т.п.	<input type="checkbox"/>
02.01.2003	23500	12	Иное	<input checked="" type="checkbox"/>
02.01.2003	16500	12	Покупка товара	<input type="checkbox"/>
02.01.2003	46500	36	Покупка товара	<input checked="" type="checkbox"/>
02.01.2003	58000	48	Покупка и ремонт недвижимости	<input type="checkbox"/>
02.01.2003	58500	42	Покупка товара	<input type="checkbox"/>
02.01.2003	20500	12	Покупка товара	<input checked="" type="checkbox"/>
02.01.2003	3500	6	Покупка и ремонт недвижимости	<input checked="" type="checkbox"/>
03.01.2003	27500	12	Покупка и ремонт недвижимости	<input checked="" type="checkbox"/>

В таблице каждое поле набора данных размещается в отдельном столбце. Столбцы озаглавлены метками полей, а если метка не была задана, то именами полей. Ширину и порядок столбцов можно менять при помощи мыши.











В таблице можно настроить объединение заголовков столбцов. Например, есть два заголовка Продажи Сумма и Продажи Количество. Если переименовать (например, с помощью обработчика **Настройка набора данных**) метку первого столбца в Продажи|Сумма, а второй – в Продажи|Количество, то получим объединение заголовка в шапке таблицы.

Продажи	
Сумма	Количество


Символ «|» подсказывает визуализатору место в слове, где заканчивается общее название у двух заголовков.

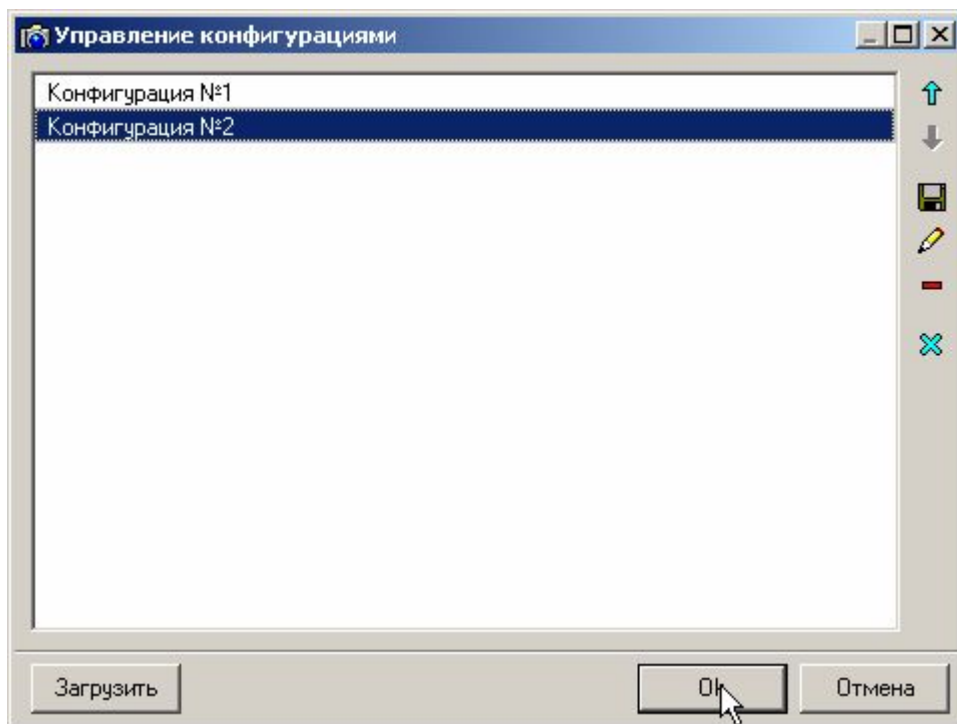
Последовательное нажатие левой кнопкой мыши по заголовку активирует сортировку по данному столбцу в следующем порядке: сортировка по возрастанию  – сортировка по убыванию  – исходное состояние. Столбцы логического типа показываются в виде флажков.

В верхней части окна таблицы представлена панель инструментов, кнопки которой открывают доступ к следующим функциям.

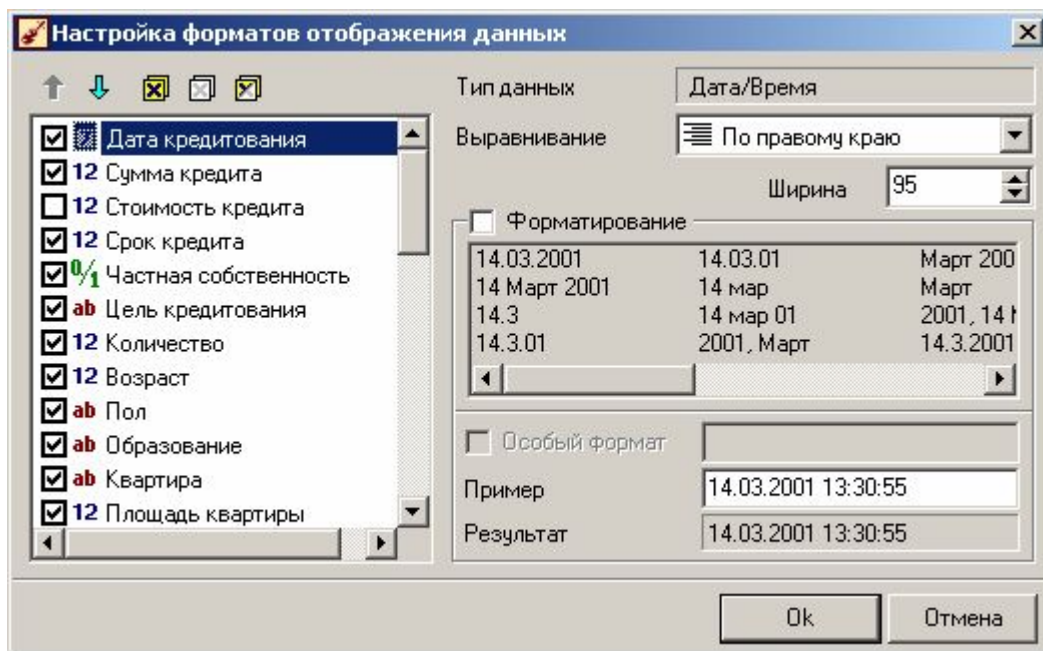
	Функция	Горячая клавиша	Описание
	<i>управление конфигурациями</i>	–	Сохранение и восстановление конфигураций отображения таблицы
	<i>настройка полей</i>	<F11>	Позволяет настраивать видимость полей, отображаемых в таблице, а также задавать их формат и способ выравнивания
	<i>способ отображения</i>	<Ctrl+F12>	Переключение между отображением данных в виде таблицы или в виде формы
	<i>статистика</i>	–	Позволяет посмотреть статистику по текущим данным таблицы. Аналогично визуализатору <i>Статистика</i> , но открывается внизу таблицы, а не в отдельном окне
	<i>фильтрация</i>	<Ctrl+D>	Позволяет выполнять фильтрацию записей в таблице по заданным условиям
	<i>первая запись</i>	<Ctrl+PgUp>	Переход на первую запись набора данных
	<i>предыдущая запись</i>	<PgUp>	Переход на предыдущую запись набора данных
28 / 149	<i>номер строки</i>	–	Индикатор текущей записи
	<i>следующая запись</i>	<PgDn>	Переход на следующую запись набора данных
	<i>последняя запись</i>	<Ctrl+PgDn>	Переход на последнюю запись набора данных
	<i>экспорт</i>	–	Вызывается окно выбора файла для экспорта данных из таблицы в один из доступных текстовых форматов: <i>MS Excel</i> , <i>RTF</i> , <i>HTML</i> , <i>TXT</i> , <i>CSV</i> . По умолчанию предлагается экспорт в <i>MS Excel</i> . В версии Academic доступны не все форматы экспорта.


Однажды настроенный вид таблицы (к примеру, с различными фильтрами, форматами и видимостью столбцов и т.п.) можно сохранить, чтобы впоследствии быстро вернуться к нему.

Для этого в раскрывающемся по кнопке  списке нужно выбрать пункт **Сохранить конфигурацию...** и далее ввести ее название. Загрузить новую конфигурацию, можно, выбрав ее из списка конфигураций.




При вызове настройки полей появляется соответствующее диалоговое окно. В нем можно скрыть или сделать видимыми различные поля таблицы, определить способ выравнивания содержимого, ширину поля, а также задать формат отображения числовых данных и дат.



Кнопка  переключает способ отображения набора данных, который может быть не только табличным, но и в виде формы. Это удобно, когда набор данных содержит большое количество столбцов.

Кнопка  открывает окно настройки условий фильтрации на набор данных.

При включенном фильтре цвет кнопки меняется на , а цвет заголовков столбцов, которые участвуют в фильтре, изменяется на красный:

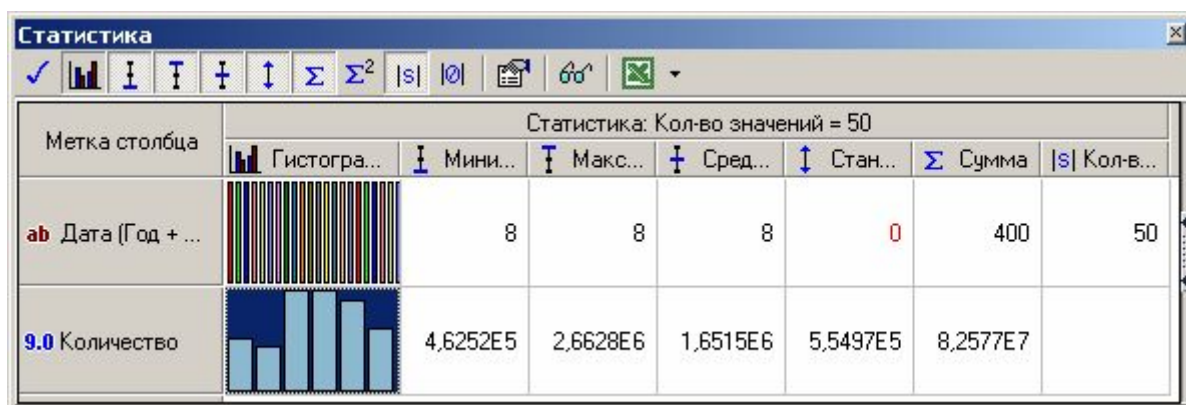
Дата (Год + Месяц)	Количество
2002-M01	355000
2002-M02	340000
2002-M03	405000
2002-M04	452000
2002-M05	464000
2002-M06	437000

Кнопка  открывает визуализатор **Статистика**, но не в отдельном вкладке, а в нижней части визуализатора **Таблица**.

Визуализатор *Статистика*

Статистика служит для отображения основных статистических характеристик набора данных конкретного узла.


Статистические характеристики отображаются в таблице по каждому полю выборки. В верхней части окна статистики отображается общее количество записей в наборе данных. Панель инструментов окна статистики позволяет управлять отображением статистических характеристик (среднее, минимум, максимум и т.п.) с помощью группы кнопок



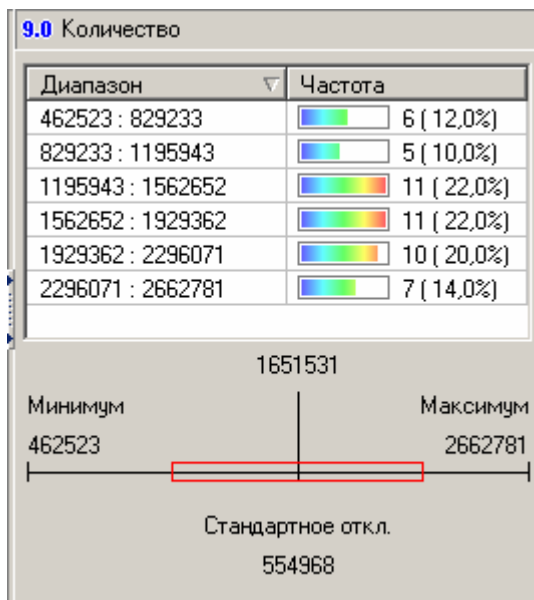
Для полей дискретного типа, кроме прочих, всегда рассчитываются следующие статистические показатели:

- количество уникальных значений,
- количество пустых значений.

Просмотреть список уникальных значений можно следующими способами:

- двойной щелчок по ячейке **Количество уникальных значений** или по ячейке **Гистограмма**,
- кнопка  **Обзор статистики**.

Для поля непрерывного типа в обзоре статистики строится гистограмма распределения частот, она же в уменьшенном виде всегда показывается в соответствующем столбце.



Визуализатор Сведения

Визуализатор **Сведения** позволяет просмотреть все параметры, с которыми был выполнен тот или иной процесс преобразования данных, в результате которого была сформирована новая выборка: импорт, обработка одним из методов или экспорт. Такими параметрами являются время и длительность выполняемого процесса, условия остановки, наличие первичного ключа, ограничители столбцов, разделители целой и дробной частей чисел, элементов даты и т.д.

Предусмотрено два вида представления описания: в виде дерева и текстовый. По умолчанию устанавливается вид дерева.

[-] Узел	
Имя	146
Метка	Данные по продажам
Описание	
[-] Объект	Текстовый файл (..\Samples\TradeSales.txt)
Максимальное время выполнения	0
Время выполнения (мс)	31
Начало процесса	2007.09.03 11:31:45
Конец процесса	2007.09.03 11:31:46
Время выполнения	0:00:00
Процесс остановлен по условию останова	False
Процесс остановлен пользователем	False
Текстовый файл	..\Samples\TradeSales.txt
Добавить первичный ключ	False
Разделитель столбцов	Табуляция
Ограничитель строк	"
Считать последовательные разделители одним	False

Визуализатор в основном предназначен для оперативного анализа текущих настроек узлов и для поиска возможных ошибок.

Визуализатор **Сведения** является единственно доступным для узлов экспорта.

Практическая работа:

- 1 Откройте проект Deductor, созданный на прошлом занятии. Настройте следующие визуализаторы к любому узлу импорта: **Таблица**, **Статистика**. Перейдите в режим формы и обратно. Имеются ли пропуски в записях?
- 2 В визуализаторе **Таблица** настройте, чтобы при отображении к значениям в Поле3 добавлялось слово «кг.». Сохраните конфигурацию визуализатора под названием К1.
- 3 Сделайте первые три столбца невидимыми. Сохраните конфигурацию визуализатора под названием К2.
- 4 Вернитесь к конфигурации К1.
- 5 В визуализаторе **Таблица** установите фильтр Поле6 = не пустой. Удалите фильтр.

Вопросы для проверки:

- 1 Какие характеристики набора данных показывает визуализатор **Статистика**?
- 2 Что означает красный заголовок столбца в визуализаторе **Таблица**?
- 3 Как обнаружить, имеются ли в столбце пропущенные значения?
- 4 Для чего предназначен визуализатор **Сведения**?
- 5 Как скрыть столбец в визуализаторе **Таблица**?
- 6 К существующему в сценарии узлу импорта необходимо добавить еще один визуализатор. Что предпринять?

Занятие 5. Узлы *Сортировка*, *Замена* и *Фильтрация*

Сортировка

Обработчик **Сортировка** предназначен для изменения порядка следования записей в наборе данных в соответствии с выбранным типом сортировки.

Результатом выполнения сортировки является новый набор данных, записи в котором следуют в соответствие с заданными параметрами сортировки.

Если сортировка производится по одному полю, то все записи исходного набора данных располагаются в порядке возрастания или убывания его значений. Если сортировка производится по двум или более полям, то действуют следующие правила:

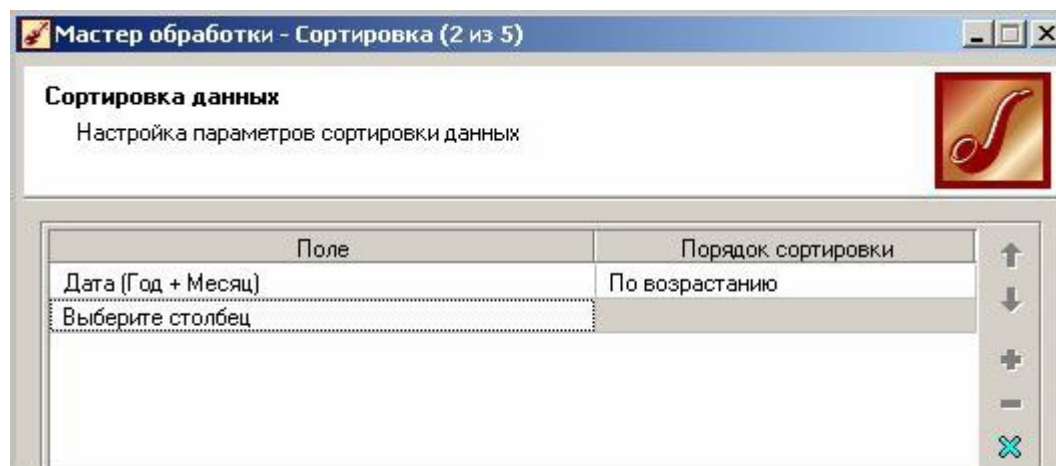
- 1 Сначала записи сортируются в заданном порядке для первого поля.
- 2 В каждом наборе одинаковых значений первого поля записи располагаются в заданном порядке для второго поля.

И так далее для всех полей, подлежащих сортировке.

Обработчик **Сортировка** находится в группе узлов **Трансформация данных** мастера обработки.

В единственном окне настройки параметров сортировки мастера обработки представлен список условий сортировки, в котором содержатся две графы:

- Имя поля – содержит имена полей, по которым следует выполнить сортировку.
- Порядок сортировки – содержит порядок сортировки данных в соответствующем поле – по возрастанию или по убыванию.



Замена данных



Обработчик **Замена данных** предназначен для замены значений набора данных по таблице подстановок, которая содержит пары, состоящие из исходного значения и результирующего значения.

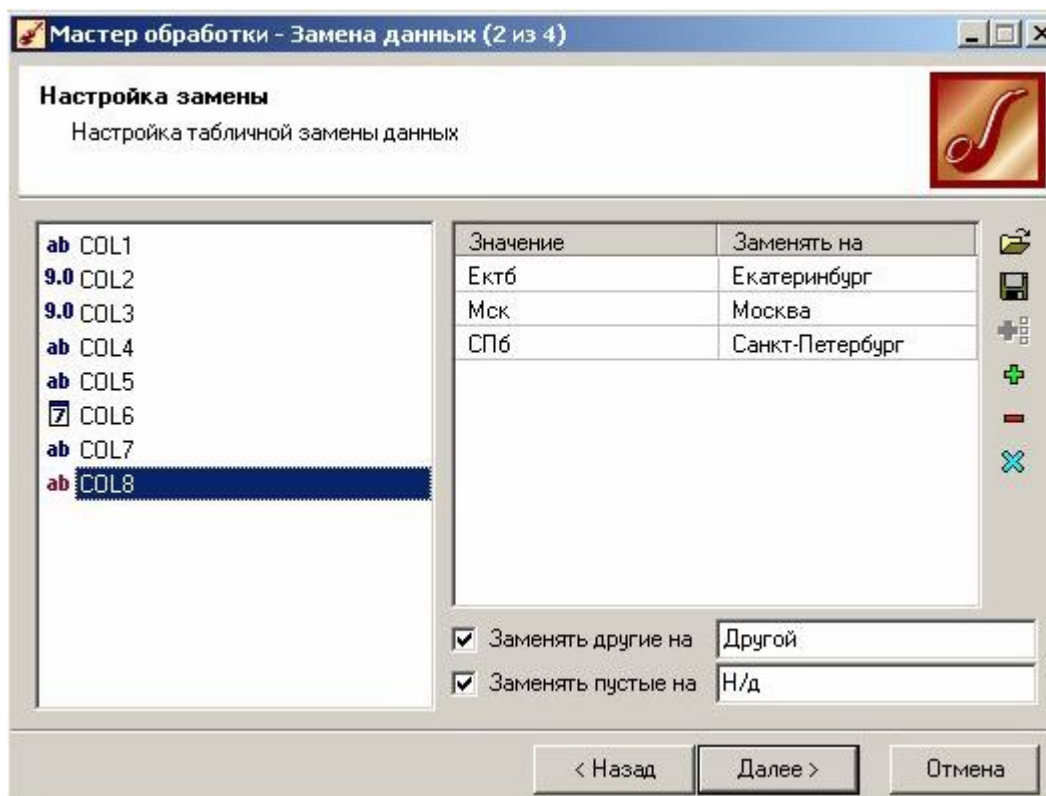
Пример таблицы подстановок.

Значение	Заменять на
Мск	Москва
СПб	Санкт-Петербург
Ектб	Екатеринбург


Для каждого значения исходного набора данных ищется соответствие среди исходных значений таблицы подстановки. Если соответствие найдено, то значение меняется на соответствующее выходное значение из таблицы подстановки. Если значение не найдено в таблице, оно может быть либо заменено значением, указанным для замены «по умолчанию», либо оставлено без изменений (если такое значение не указано).

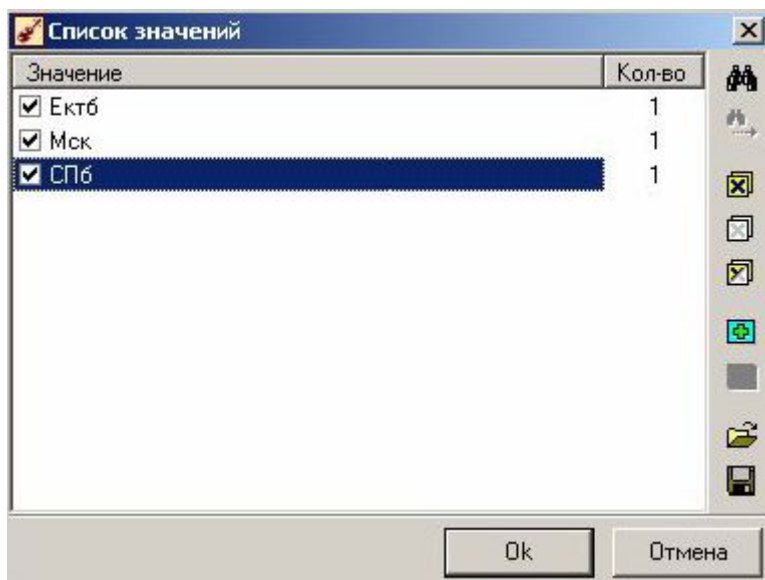
В результате замены для каждого поля, которое в нем участвует, создается новое поле с префиксом `_REPLACE` как к имени, так и к метке поля. Например, для поля Город после узла **Замена данных** появится новое поле `Город_REPLACE`.

Обработчик **Замена данных** находится в группе узлов **Трансформация данных** мастера обработки. В окне настройки параметров замены для каждого поля можно ввести таблицу подстановок. Добавление новой строки в таблицу подстановок производится нажатием кнопки , удаление существующей – .



В таблице подстановок должны быть заполнены два поля:


- **Значение** – заменяемое значение поля исходной таблицы. Если поле дискретное, то для ввода значения можно воспользоваться кнопкой выбора , где флажками отметить нужные значения. При этом откроется диалоговое окно:




- **Заменять на** – значение для замены того, что указано в поле Значение.

Внизу таблицы подстановок расположены еще два флага, которые при необходимости можно включить:

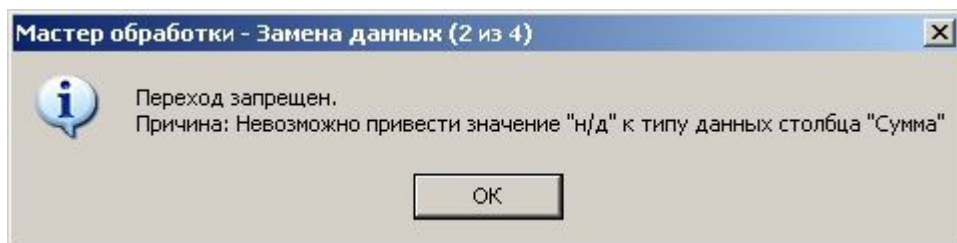
- **Заменять другие на** – на какое значение следует заменить значения, не указанные в таблице замены. Для этого установите флажок и в поле напротив введите значение для замены.
- **Заменять пустые на** – на какое значение заменять пустые значения поля.

Таблицу подстановки, кроме непосредственного ввода, можно заполнить, загрузив ее из текстового файла (кнопка ). Формат текстового файла должен быть следующим:

<заменяемое значение><символ табуляции><значение для замены>

И наоборот, список подстановок можно сохранить в текстовый файл (кнопка ).

Если по полю настроена таблица подстановок, иконка типа данных меняет свой цвет на **красный**. Попытка сделать замену данных с числового типа на строковый может потерпеть неудачу с выдачей соответствующего сообщения. Например, заменить все пустые значения на «н/д» в вещественном поле Сумма не получится, т.к. поле уже становится не вещественным, а строковым. Поэтому предварительно необходимо преобразовать поле Сумма в строковый тип при помощи обработчика **Настройка набора данных**.




Фильтрация

Обработчик **Фильтрация** предназначен для исключения из набора данных записей, не удовлетворяющих условиям фильтрации.

Обработчик **Фильтрация** находится в группе узлов **Очистка данных** мастера обработки.

Параметры фильтрации задаются в виде списка условий, который содержит следующие столбцы.

- 1** Операция – позволяет установить функцию отношения «И» или «ИЛИ» между полями, для каждого из которых выполняется фильтрация. Возможна фильтрация по нескольким условиям для нескольких полей одновременно. В результате фильтрации по каждому из полей или условий будет получено отдельное множество значений. Функция в поле Операция устанавливает отношение между этими множествами. Если используется отношение «И», то в результирующий набор будут включены записи, удовлетворяющие условиям фильтрации по обоим полям. Если используется отношение «ИЛИ», то в выходной набор будут включены данные, удовлетворяющие хотя бы одному из условий. Установка отношений возможна, только если настроены два или более условия фильтрации. Для выбора операции следует дважды щелкнуть левой кнопкой мыши в столбце Операция для соответствующего условия и из списка, открываемого кнопкой, выбрать нужную функцию отношения. По умолчанию устанавливается отношение «И».
- 2** Имя поля – позволяет выбрать поле, по значениям которого должна быть выполнена фильтрация. Одно и то же поле может быть использовано в нескольких условиях.
- 3** Условие – указывается условие, по которому нужно выполнить фильтрацию для данного поля.

Для выбора условия достаточно дважды щелкнуть мышью в соответствующей ячейке и в списке условий, открываемом кнопкой , выделить нужное условие. Доступны следующие условия фильтрации:

- (равно), < (меньше), <= (меньше или равно), > (больше), >= (больше или равно), <> (не равно) – отбираются только те записи, значения которых в данном поле удовлетворяют заданному выражению;
- пустой – отбираются только те записи, для которых в данном поле содержится пустое значение. В этом случае поле Значение не используется;
- не пустой – отбираются только те записи, для которых в данном поле не содержится пустое значение. В этом случае поле Значение не используется;
- содержит – отображаются только те записи, которые в данном столбце содержат указанное значение;
- не содержит – отображаются только те записи, которые в данном столбце не содержат указанное значение;
- в интервале, вне интервала – для числовых полей и полей типа Дата/время отбираются только те записи, значения которых в данном столбце лежат в выбранном диапазоне (вне выбранного диапазона);
- в списке, вне списка – отбираются только те записи, которые содержатся в выбранном списке (вне выбранного списка);
- начинается на, не начинается на – для строковых полей отбираются записи, значения которых в данном столбце начинаются (не начинаются) на введенную последовательность символов.
- заканчивается на, не заканчивается на – для строковых полей отбираются записи, значения которых в данном столбце заканчиваются (не заканчиваются) на введенную последовательность символов.
- первый, не первый – для полей типа **Дата/время** – по данному полю отбираются первые (не первые) N периодов от выбранной даты. Периодом может быть день, неделя, месяц, квартал, год. Например, если выбрать условие **первые 3 дня от 29.11.2004**, то будут отобраны записи, в которых значение данного поля равно 29.11.2004, 30.11.2004, 01.12.2004 – 3 последующих дня.


- последний, не последний – для полей типа **Дата/время** отбираются последние (не последние) N периодов от выбранной даты. Периодом может быть день, неделя, месяц, квартал, год. Например, если выбрать условие *последние 3 дня* от 29.11.2004, то будут отобраны записи, в которых значение данного поля равно 29.11.2004, 28.11.2004, 27.11.2004 – 3 предыдущих дня.
- 4** Значение – указывается значение(я), по которому будет производиться фильтрация записей в соответствии с заданным условием. Способ ввода значения будет различным в зависимости от типа данных и выбранного условия. Допустим, в качестве условия выбрана операция отношения «=», «<>», «>» и т.д. Если данные в поле являются непрерывными (т.е. числовыми), то достаточно дважды щелкнуть мышью в соответствующей ячейке, чтобы появился курсор, затем ввести значение (число). Если поле, по которому выполняется фильтрация, имеет тип «строка» (т.е. является дискретным), то в результате двойного щелчка в столбце Значение появится кнопка выбора, которая откроет окно «Список уникальных значений», где будут отображены все уникальные значения поля и их количество. Чтобы выбрать значение для условия отбора, достаточно выделить его и щелкнуть **Ok**, либо просто дважды щелкнуть мышкой на нужном значении. Если выбрано условие между или не между, тогда после щелчка мышки откроется окно, в котором необходимо указать верхнюю и нижнюю границы интервала, и так далее.

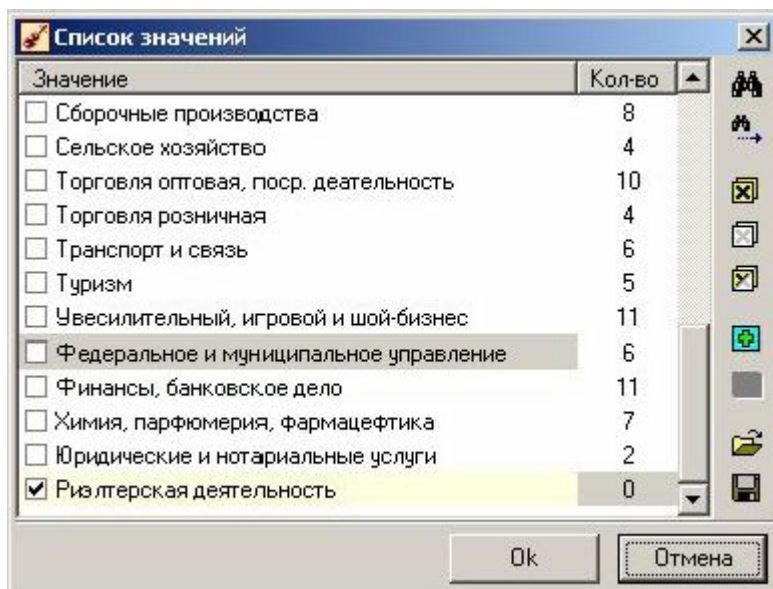
Флажок **Учитывать регистр** учитывает регистр символов при отборе.

Внизу окна настроек в автоматическом режиме формируется выражение для фильтрации, полученное объединением всех условий, например:

([Размер ссуды, руб] в интервале [2000..5000]) И ([Цель ссуды] = 'Покупка товара').

Иногда возникает необходимость построить фильтр для дискретного поля с условием *в списке, вне списка* для значений, которые не существуют в наборе данных (но предполагается, что они

могут появиться в будущем). Выходом служит кнопка  **Добавить значение** в окне выбора списка значений. Количество записей такого «несуществующего» списочного значения всегда будет равно нулю, а строка – подкрашена светло-желтым цветом.



Практическая работа:

- 1** Создайте новый проект. Импортируйте в него текстовый файл **CreditSample.txt**, идущий в поставке Deductor (по умолчанию расположен в каталоге /Samples директории установки Deductor).

- 2 Отсортируйте этот набор данных по следующим полям в порядке возрастания: Срок ссуды, Размер ссуды, Количество иждивенцев.
- 3 Сделайте следующую замену (после **Сортировки**) в поле Семейное положение: значение Да измените на Женат/замужем, Нет – на Холост/Не замужем.
- 4 Сделайте следующую замену (после предыдущего узла **Замена данных**) в поле Количество иждивенцев: значение 0 – на Нет, 1 – без изменений, 2 и 3 – 2 и более. Используйте два способа – непосредственным вводом в мастере обработки и через файл таблицы соответствий. Файл подстановок предварительно создайте в любом текстовом редакторе, например, в Блокноте.
- 5 Старое поле Количество иждивенцев удалите из набора данных, а новое поле Количество иждивенцев_REPLACE переименуйте в Иждивенцы.
- 6 Отфильтруйте набор данных, полученный в п. 5 по полю Иждивенцы так, чтобы в выходной набор попали только строки, у которых значение в поле Иждивенцы не равно Нет. Сколько записей прошло через фильтр?
- 7 Отфильтруйте набор данных, полученный в п. 5 по полю Иждивенцы так, чтобы в выходной набор попали только строки, у которых значение в поле Иждивенцы не равно Н/д. Сколько записей прошло через фильтр?
- 8 Продолжите фильтровать набор данных, полученный в п. 6. Наложите следующий фильтр, в который попадают все записи, удовлетворяющие условиям а либо условиям б:
 - а. Размер ссуды – от 2000 до 5000, Цель ссуды – Покупка товара.
 - б. Цель ссуды – Иное.
- 9 Сколько записей прошло через фильтр?
- 10 Отсортируйте последний набор данных по полю Код.

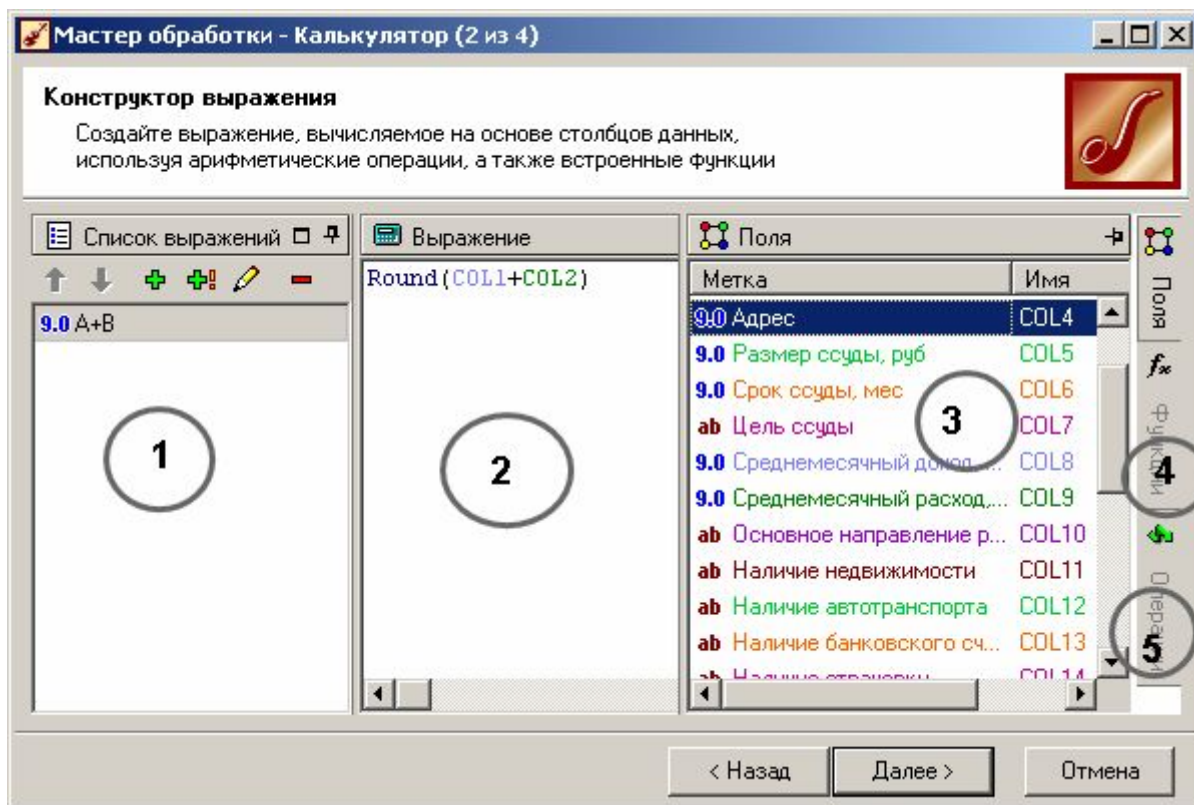
Вопросы для проверки:

- 1 Как работает обработчик **Сортировка**?
- 2 Можно ли отсортировать набор данных по нескольким полям?
- 3 Для чего предназначен узел **Замена данных**?
- 4 Как определить в мастере обработки, что для поля настроена замена?
- 5 Как работает **Замена данных**?
- 6 Какие существуют способы заполнить таблицу подстановок?
- 7 Для чего предназначен узел **Фильтр**?
- 8 Какие условия фильтрации существуют?
- 9 Сколько записей будет отфильтровано в результате фильтра «([Размер ссуды, руб] в интервале [2000..5000]) И ([Цель ссуды] = 'Покупка товара') И ([Цель ссуды] = 'Иное')»?
- 10 Что делать, если нужно поставить фильтр по значению, которого в данный момент нет в рассматриваемом наборе данных?

Занятие 6. Узел *Калькулятор*




Калькулятор предназначен для добавления в набор данных новых полей, которые рассчитываются по определенным правилам на основе столбцов данных и встроенных функций.


Обработчик **Калькулятор** находится в группе узлов **Прочее** мастера обработки. Вся настройка осуществляется в окне мастера **Конструктор выражения**.



- 1 – Область списка вычисляемых выражений. Каждое вычисляемое выражение будет новым столбцом в результирующем наборе данных.
- 2 – Формула, по которой будет рассчитываться выражение (окно выражения).
- 3 – Список всех существующих столбцов текущего набора данных, состоящих из имен и меток. Для каждого столбца показывается имя и метка.
- 4 – Открывает вкладку со списком встроенных функций.
- 5 – Открывает вкладку со списком доступных арифметических, логических и других операций.

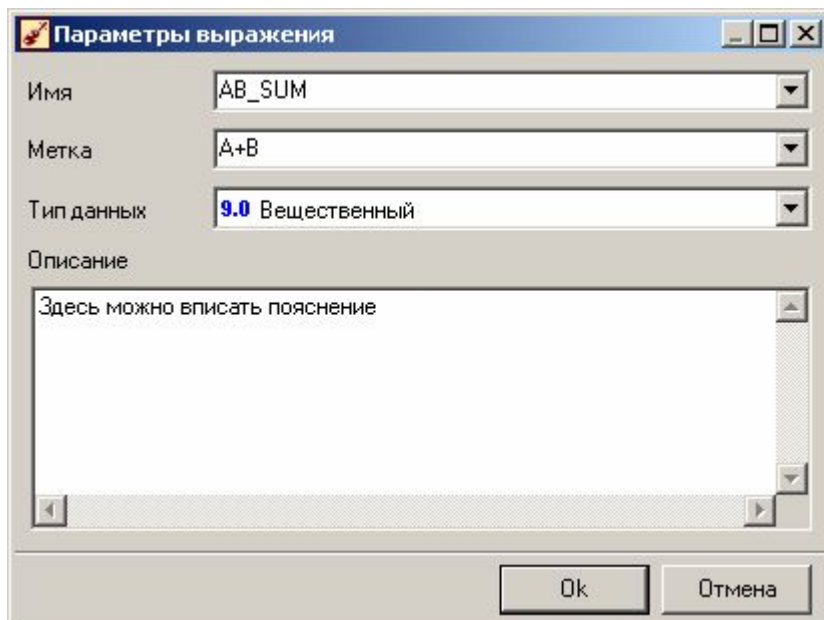
Область списка вычисляемых выражений изначально содержит одно пустое выражение. Для управления списком вычисляемых выражений предусмотрены следующие кнопки:

-  (**Ctrl+Up**) – Переместить текущее выражение на одну позицию вверх по списку.
-  (**Ctrl+Down**) – Переместить текущее выражение на одну позицию вниз по списку.
-  (**Num+**) – Добавить новое выражение с параметрами, устанавливаемыми по умолчанию, и пустой формулой.

 – Добавить новое выражение с типом данных, описанием и формулой как у текущего выражения.

 (**Num-**) – Удалить текущее выражение.

Двойным щелчком мыши на имени выражения в списке вызывается **Диалог редактирования параметров выражения**.



- Имя – строка, которая будет служить идентификатором столбца в процедурах обработки. Может состоять только из латинских символов и должно быть уникальным в пределах одного набора данных.
- Метка – метка нового столбца. Именно она отображается в списке вычисляемых выражений. Уникальность меток не требуется.
- Тип данных – тип данных вычисляемого выражения. Тип выбирается из списка, открываемого щелчком по кнопке в правой части поля.
- Описание – произвольная информация, описывающая вычисляемое выражение.

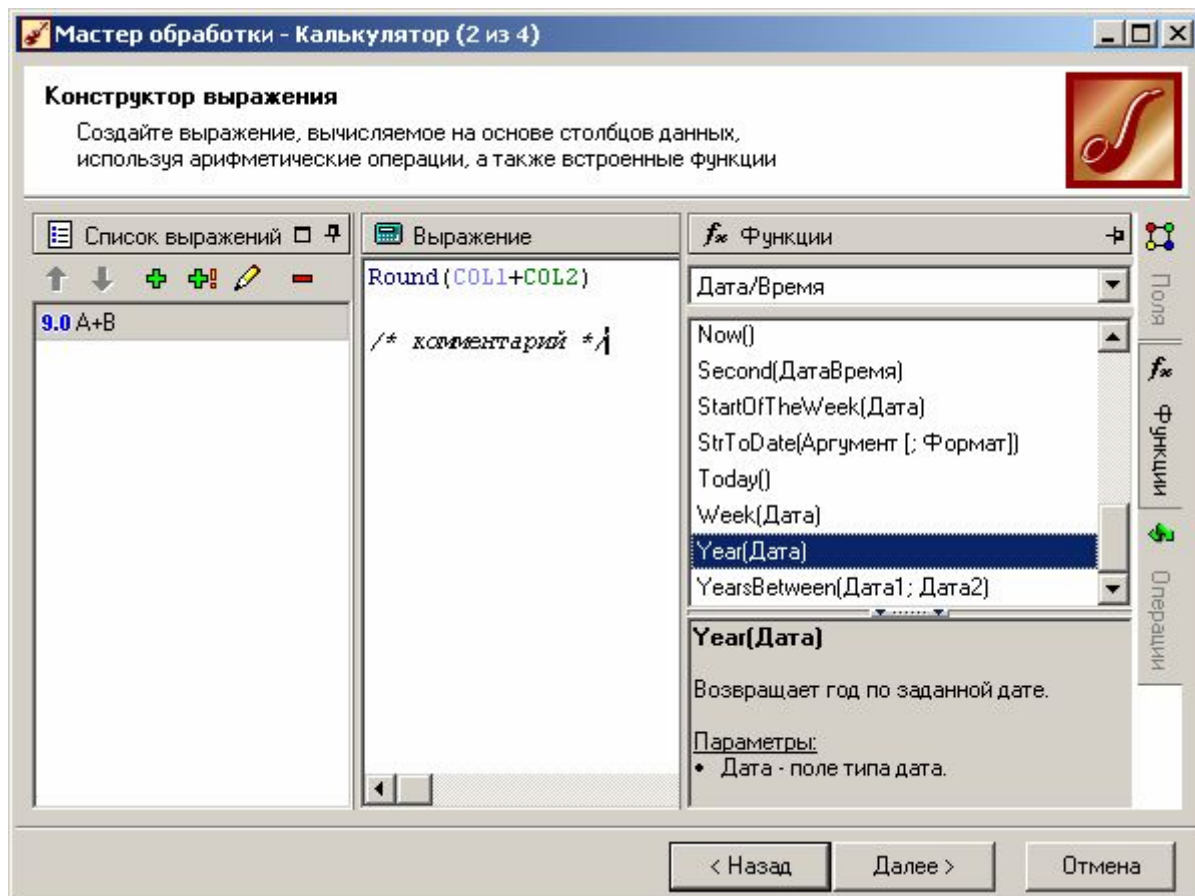
Изначально при открытии страницы **Конструктора** список выражений содержит только один новый столбец. По умолчанию для нового выражения назначается метка `Выражение_N`, где N – номер, обеспечивающий уникальность. Имена полей, формируемых в результате вычислений по данному выражению, назначаются автоматически и имеют вид: `EXPR_N`.

Далее требуется ввести рассчитываемую формулу в окно выражений. Правила составления выражений соответствуют общепринятым в математике, в частности, число открывающих скобок должно равняться числу закрывающих. Выражение может содержать:

- Числа в явном виде.
- Переменные в виде имен столбцов.
- Скобки, определяющие порядок выполнения операций.
- Знаки математических операций и отношений.
- Имена функций.
- Даты в формате ДД.ММ.ГГ, обязательно указываемые в кавычках. Такой способ ввода даты, хотя и допускается, но может оказаться не переносимым между разными компьютерами. По этой причине лучше использовать функцию **STRTODATE()**.

- Строковые выражения в двойных кавычках: "строковое выражение".
- Однострочные и многострочные комментарии. Однострочный комментарий начинается символами // (два слеша) и продолжается до конца строки. Многострочным комментарием считаются все символы, содержащиеся между скобками /* и */ (звездочка-слеш).

Выражение можно ввести вручную с клавиатуры, однако удобнее выбирать функции, переменные и знаки операций с помощью мыши. Для добавления в формулу функций следует справа выбрать вкладку **Функции**. Все функции в ней сгруппированы по видам.



Чтобы ввести функцию в выражение, достаточно дважды щелкнуть по ее имени в списке, либо, удерживая, перетащить ее мышью в нужную область формулы. Имя функции в выражении появляется вместе со скобками, куда следует ввести аргумент или аргументы. Аргументами могут быть числа в явном виде, строки в кавычках, даты в кавычках, имена функций, имена полей, а также арифметические, логические и строковые выражения. Имена полей удобно вводить с помощью двойного щелчка в списке полей. Если в аргументе несколько полей, то их имена разделяются точкой с запятой.

В окне ввода выражения можно вывести подсказку – комбинация клавиш **Ctrl+пробел**.

При создании формул при разработке сценариев очень часто используются функции **IF** и **IFF**.

Функция	Описание
IF(Условие; Значение1; Значение2)	Возвращает Значение1, если Условие истинно или Значение2, если Условие ложно. Результат функции имеет строковый тип.
IFF(Условие; Значение1; Значение2)	Возвращает Значение1, если Условие истинно или Значение2, если ложно. Результат функции может иметь любой тип.

В том случае, когда нужно создать два новых столбца Поле1 и Поле2, а Поле2 рассчитывается на основе Поля1, необходимо создать два узла типа **Калькулятор**.

Особенность работы узла при возникновении ошибок

Создание нового поля при помощи **Калькулятора** на каком-либо наборе данных не означает, что в последствии не возникнут ошибки при расчете значений. Например, формула имела вид Поле1/Поле2. Что будет, если в Поле2 окажется нулевое или пустое значение? Узел **Калькулятор** имеет следующее правило работы в таких ситуациях: при возникновении любой ошибки в расчете значения записи в рассчитываемое поле заносится значение NULL (пустое значение) и сообщение об ошибке не выдается. Это нужно учитывать при разработке и отладке сценариев. В случае, когда формула в **Калькуляторе** ссылается на несуществующий столбец, то будет выдано сообщение типа

«Столбец "Имя" ("Название") должен существовать в исходном источнике данных»

и узел не будет выполнен. Такое может случиться, например, когда набор данных, находящийся над узлом **Калькулятор**, поменял свою структуру или имена полей.

Практическая работа:

- 1 Создайте новый проект. Импортируйте в него текстовый файл **CreditSample.txt**, идущий в поставке **Deductor** (по умолчанию расположен в каталоге /Samples директории установки **Deductor**).
- 2 Создайте новое поле Дата обработки, значения в котором равны текущей дате.
- 3 Создайте новое поле Размер ссуды у.е., который рассчитывается делением на 30 поля Размер ссуды, руб. Все значения в новом поле должны быть округлены до второго знака.
- 4 Создайте новое поле Флаг, значение в котором истинно, если выполняется условие:

Среднемесячный доход > 2000 и Наличие недвижимости = Да.

- 5 Создайте еще один столбец, значение в котором равно 1, если выполняется условие:

Флаг = TRUE и Давать кредит = FALSE.

- 6 Создайте новое поле RATE, в котором хранится значение в поле Срок ссуды, возведенное в степень 0,6.
- 7 Создайте новое поле Сегмент, которое делит всех заемщиков на сегменты по следующим правилам (используйте функцию **IF/IFF**):
 - 1) ЕСЛИ Возраст >= 50 и Среднемесячный доход < 6000 ТО Сегмент = Сегмент 1
 - 2) ЕСЛИ Возраст < 30 и ТО Сегмент = Сегмент 2
 - 3) Сегмент = Сегмент 3 во всех остальных случаях, не удовлетворяющим п. 1) и 2).

Вопросы для проверки:

- 1 Для чего предназначен обработчик **Калькулятор**?
- 2 Как добавить новый столбец?
- 3 Какой символ используется для разделения параметров в функциях калькулятора?
- 4 Как ввести формулу для расчета значений столбца?
- 5 Как вывести подсказку для функции в окне создания выражений?
- 6 Чем отличаются функции **IF** и **IFF**?
- 7 Что делает функция **ISNULL**?
- 8 Как добавить существующее имя поля в формулу?
- 9 Как посмотреть описание той или иной функции?
- 10 Что делают следующие функции: **NOW()**, **TODAY()**, **ROUND()**, **POW()**?
- 11 Что будет, если в **Калькуляторе** создать новый столбец вещественного типа и написать для него формулу $15/0$?

Занятие 7. Использование скриптов

Введение

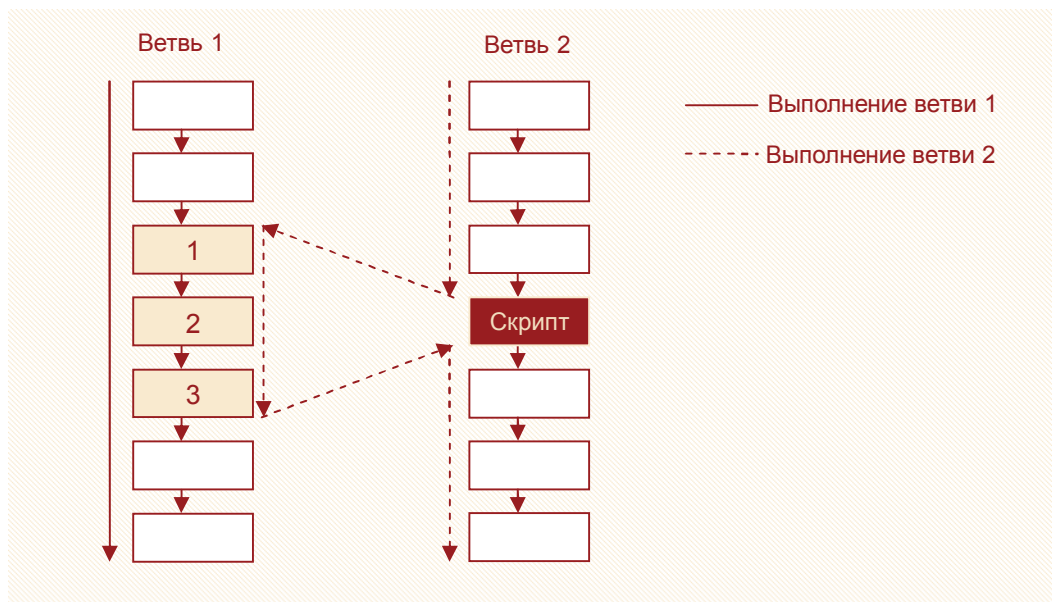
Скрипты предназначены для автоматизации процесса добавления в сценарий однотипных ветвей обработки. Это нужно в следующих случаях:

- требуется выполнить часть сценария (т.е. последовательность узлов) на другом наборе данных;
- требуется применить модель (дерево решений, нейронная сеть) на новых данных.

Если повторное выполнение части сценария можно обойти, используя копирование веток, то в случае применения аналитической модели к новым данным без обработчика **Скрипт** обойтись невозможно.

По сути скрипт представляет собой динамическую копию выбранного участка сценария. Скрипт является готовой частью сценария, и поэтому входящие в него узлы не могут быть изменены отдельно от исходной ветки сценария. Тем не менее, в скрипте отражаются все изменения, вносимые в ветку, на которую он ссылается, т.е. при переобучении или перенастройке узлов этой ветки все сделанные изменения будут внесены в работу скрипта.

Предположим, что после импорта данных из двух разных текстовых файлов требуется провести определенную предобработку (поменять названия столбцов, заменить данные, добавить несколько расчетный столбцов), а затем экспортировать полученные данные обратно. Для первой ветви (первого текстового файла) эти действия проводятся как обычно – последовательными шагами строится цепочка обработчиков. Для второго же источника (второго файла) достаточно создать узел импорта, к которому присоединить узел Скрипт, основанный на уже построенной первой ветви. В этом скрипте будут выполнены точно такие же действия, как в оригинальной ветви. На выходе скрипта ставится узел экспорта, и вторая ветвь обработки готова к использованию. Эту идею иллюстрирует рисунок ниже.



На рисунке показана схема выполнения ветви со скриптом, включающего три узла из другой ветки сценария. Сначала (до узла со скриптом) последовательно выполняются узлы второй ветки. Затем осуществляется переход на начальный узел скрипта, находящийся в Ветви 1. Далее последовательно выполняются уже узлы первой ветки, пока не будет достигнут конечный узел скрипта. После этого осуществляется возврат к Ветви 2 на следующий после

скрипта этап обработки, и выполнение продолжается. На ход выполнения первой ветви скрипт при этом не оказывает никакого влияния.

Особенность использования скрипта вместо копирования ветви заключается в том, что внесенные в главную ветвь изменения автоматически наследуются всеми скриптами, которые ссылаются на узлы главной ветви. В большинстве случаев это преимущество, однако, иногда при создании сценариев необходимо именно копирование узлов.


Аналогом скрипта является функция или процедура в языках программирования. Ветвь обработки строится один раз, а затем скриптами она тиражируется в другие места сценария.

Обработчик **Скрипт** находится в группе узлов **Прочее**.

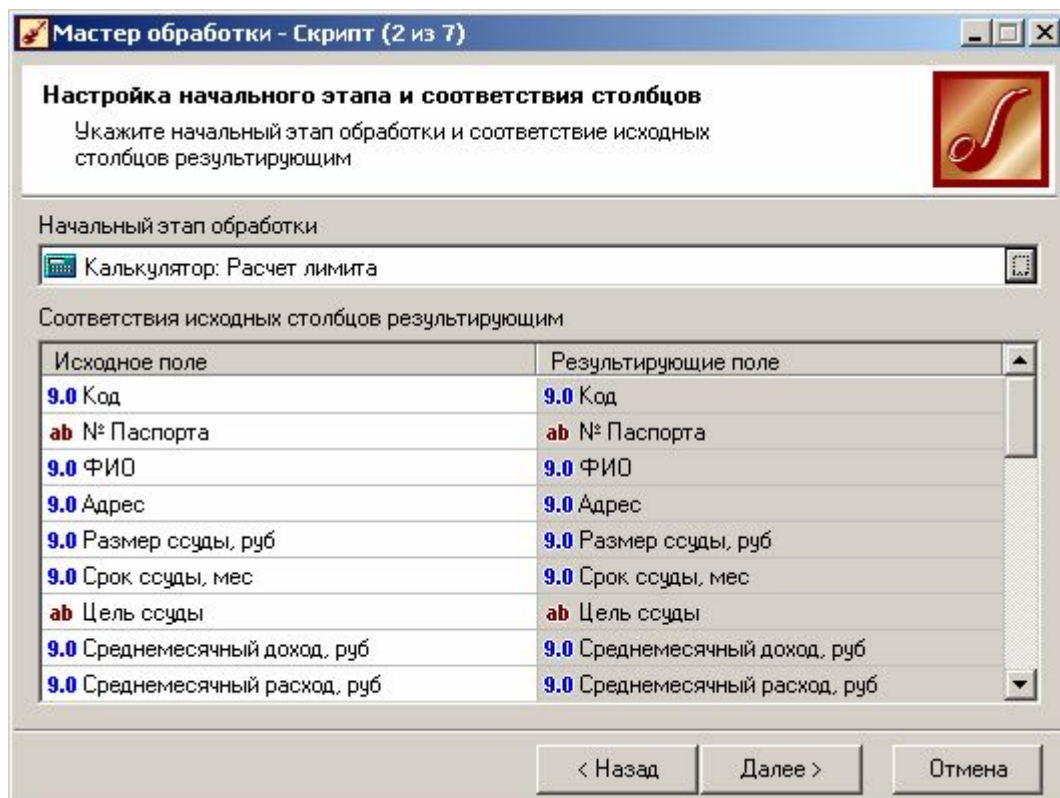
Создание и настройка скрипта

Настройка скрипта состоит из следующих шагов.

Шаг 1. Задание начального узла обработки и соответствия полей. Это осуществляется в окне **Настройка начального этапа и соответствия столбцов** мастера обработки узла **Скрипт**.

Для выбора начального узла нужно нажать кнопку , после чего на экране появится окно **Выбор узла**. В этом окне показано все дерево сценария. Кнопка **Ок** подтверждает выбор текущего узла в качестве начального узла скрипта, кнопка **Отмена** закрывает окно, не внося изменений.

При выборе начального узла существует следующее ограничение: начальным узлом может быть только узел обработчика (узел импорта или экспорта данных не может быть выбран).

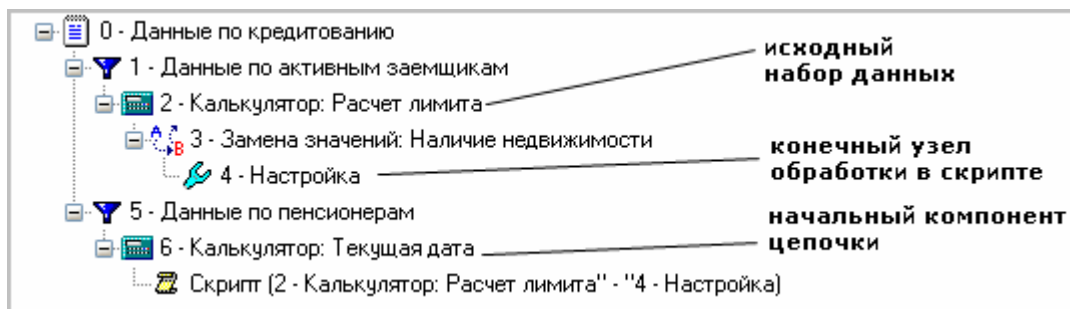


Исходное поле	Результирующее поле
9.0 Код	9.0 Код
ab № Паспорта	ab № Паспорта
9.0 ФИО	9.0 ФИО
9.0 Адрес	9.0 Адрес
9.0 Размер ссуды, руб	9.0 Размер ссуды, руб
9.0 Срок ссуды, мес	9.0 Срок ссуды, мес
ab Цель ссуды	ab Цель ссуды
9.0 Среднемесячный доход, руб	9.0 Среднемесячный доход, руб
9.0 Среднемесячный расход, руб	9.0 Среднемесячный расход, руб

В случае, когда исходный набор данных имеет меньшее число столбцов, чем начальный компонент цепочки, на экран будет выдано предупреждение:

Количество столбцов начального компонента цепочки не должно быть больше чем количество столбцов исходного набора данных.

При этом в момент обработки скрипта будет принята попытка выполнить с имеющимся набором полей. Если какое-то из отсутствующих полей является критичным для любого узла, содержащегося в скрипте, то обработка будет остановлена с выдачей сообщения об ошибке. Под исходным набором данных подразумевается тот набор данных, к которому применяется обработчик **Скрипт**, под начальным компонентом цепочки – набор данных, на который настраивается **Скрипт**.



После выбора начального узла следует задать соответствия столбцов исходного набора данных полям выбранного узла. В нижней части экрана находится таблица со списком полей исходного набора в левом столбце и полей выбранного узла – в правом. Для каждого поля начального узла надо задать поле-источник исходного набора. Для этого следует, щелкнув два раза в левом столбце напротив имени нужного поля, выбрать из выпадающего списка имя столбца входного набора.

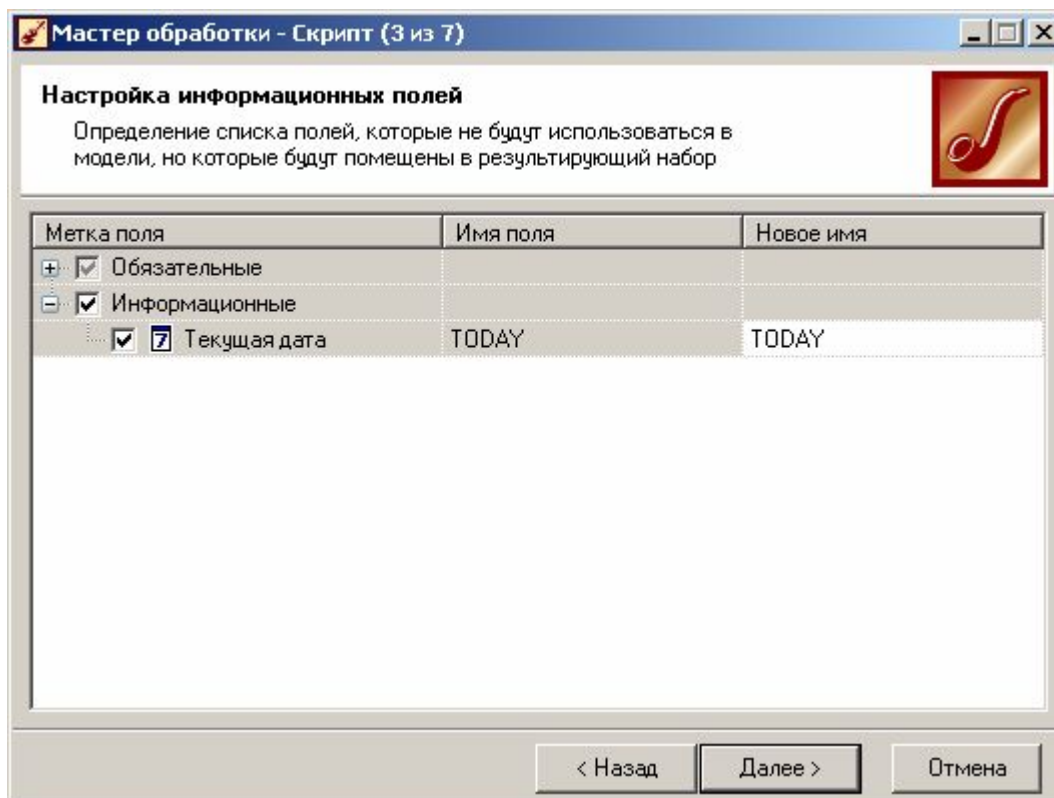
Мастер обработки узла Скрипт устроен так, что пытается автоматически сопоставить поля в источниках, совпадающие по названию и/или типу.

Настроить соответствия столбцов, которые имеют различный тип, невозможно.

Исключение имеется только для типов целый и вещественный, однако *рекомендуется всегда настраивать соответствие столбцов, имеющих одинаковый тип* (т.е. целый-целый, вещественный-вещественный).

Возможна ситуация, когда столбцам начального компонента цепочки нет сопоставимых столбцов в исходном наборе данных. В такой ситуации система выдаст следующее сообщение: «Столбцам начального компонента цепочки нельзя сопоставить столбцы исходного набора данных». При этом в момент обработки скрипта будет принята попытка выполнить с имеющимся набором полей. Если какое-то из отсутствующих полей является критичным для любого узла содержащегося в скрипте, то обработка будет остановлена с выдачей сообщения об ошибке.

Шаг 2. Этап настройки информационных полей. Это необязательный шаг мастера, который появляется в том случае, когда исходный набор данных содержит большее количество полей, чем набор данных, являющийся начальным компонентом цепочки. Под информационными полями понимаются те поля, которые не будут использоваться в скрипте, но которые будут помещены в результирующий набор данных.



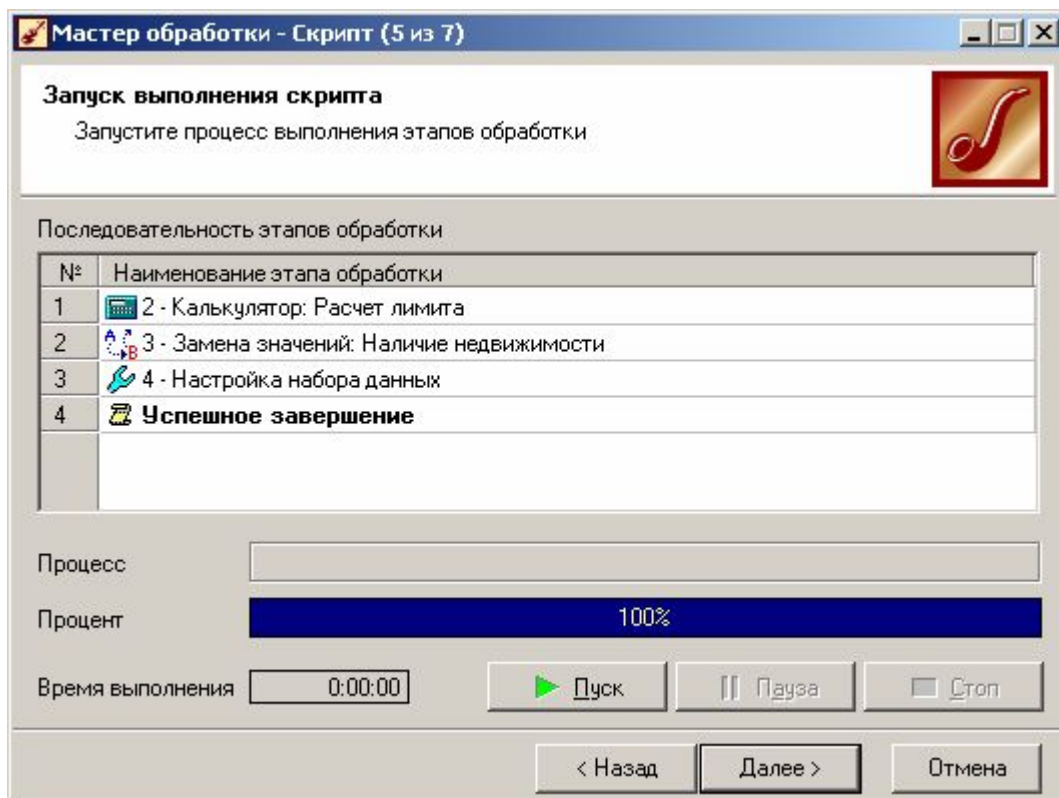
Шаг 3. Задание конечного узла обработки. Здесь существуют следующие правила.

- Начальный и конечный узлы должны находится **на одной ветви сценария**, т.е. конечный узел должен являться потомком начального узла в дереве сценария.
- Конечным узлом не может являться узел экспорта.
- На число и типы промежуточных узлов не накладывается никаких ограничений, т.е. промежуточными узлами могут быть и скрипты.

Шаг 4. Запуск процесса обработки. На данном шаге запускается собственно процесс выполнения скрипта.

В секции **Последовательность этапов обработки** показан список всех узлов, входящих в скрипт. Узлы, которые еще не выполнялись, отображаются с серыми иконками, выполненные – с цветными. Имя текущего обрабатываемого узла отображается жирным шрифтом.

Если процесс обработки остановился, это сигнализирует о возможных проблемах. Остановка может произойти в случае несоответствия типов данных алгоритму обработки, наличия в обрабатываемых полях недопустимых значений и т.д. В этом случае возможно появление окна с сообщением об ошибке. Если обработка данных была завершена успешно, то в секции **Название процесса** появится сообщение «Успешное завершение».



Практическая работа:

1. Создайте новый проект. Импортируйте в него текстовый файл **Trade.txt**, идущий в поставке Deductor (по умолчанию расположен в каталоге /Samples директории установки Deductor).
2. Добавьте после узла импорта 2-3 обработчика из изученных ранее.
3. Импортируйте в него текстовый файл **TradeSales.txt**, (он расположен там же). Добавьте к нему поле Номер строки (используйте функцию калькулятора **RowNum()**).
4. Добавьте к набору данных скрипт, выполняющий те же действия с набором данных, что и в п. 2.

Вопросы для проверки:

1. Для чего предназначен обработчик **Скрипт**?
2. В каких случаях возникает необходимость добавить в сценарий скрипт?
3. Что такое исходный набор данных, начальный и конечный узел при настройке обработчика **Скрипт**?
4. Чем отличается копирование ветви от применения скрипта?
5. Можно ли настроить соответствия столбцов, которые имеют различный тип?
6. Какие ограничения накладываются на выбор конечного узла обработки в скрипте?

Занятие 8. Групповая обработка

Узел **Групповая обработка** работает похожим на **Скрипт** образом. Основным отличием от него является то, что входной набор делится на части по указанным группам, и затем каждая группа отдельно «прогоняется» через копию цепочки узлов обработки.

Если аналогом скрипта является *процедура* в языке программирования, то аналогом групповой обработки – *цикл*. Групповая обработка позволяет создавать очень гибкие сценарии, особенно она незаменима в тех случаях, когда нужно обрабатывать отдельные «пачки» данных внутри одного набора в зависимости от статистических характеристик каждой такой «пачки» (сумма, среднее, количество записей и т.д.).

Рассмотрим групповую обработку на конкретном примере. Импортируем в **Deductor** текстовый файл **Trade.txt** (по умолчанию он расположен в каталоге /Samples). Фрагмент набора данных приведен ниже в таблице.

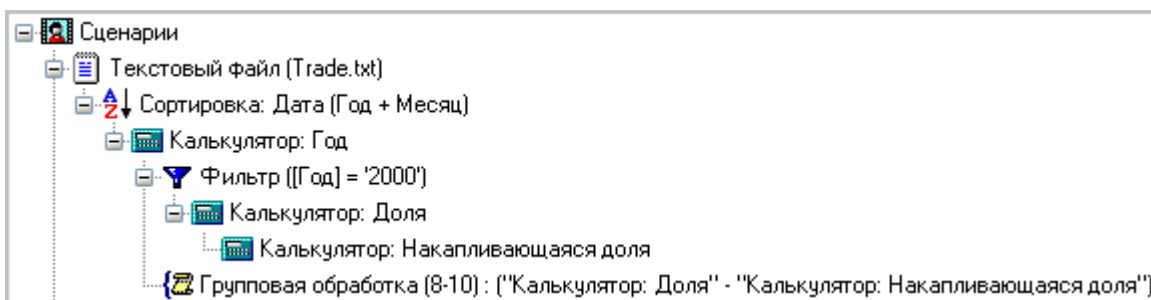
Дата (Год + Месяц)	Количество
2000-M01	462 523,419
2000-M02	633 208,196
2000-M03	660 159,299
2000-M04	617 455,341
2000-M05	597 354,479
...	...

Отсортируем его по возрастанию по полю Дата (Год + Месяц). Далее из этого поля узлом **Калькулятор** выделим год, создав новое поле с функцией

`SUBSTR(COL1;1;4).`

Пусть перед нами стоит задача: рассчитать для каждого месяца каждого года (т.е., по сути, строки набора данных) долю и долю с накоплением от годовой суммы в пределах одного года. Ситуация характеризуется тем, что у нас не один год, а несколько: с 2000 по 2004.

Воспользуемся **Групповой обработкой**. Для наглядности сначала сделаем все необходимые действия над одной группой, скажем, 2000 год, а затем «распространим» эти действия на весь исходный набор данных.



Сперва мы выделили эту группу фильтром и последовательно добавили два поля двумя калькуляторами: Доля (PART):

`ROUND(COL2/Stat("COL2";"SUM")*100;2),`

и Накапливающаяся доля (CUM_PART):

CumulativeSum("PART").

Далее добавим к исходному набору данных узел **Групповая обработка**. На первом шаге нужно указать поля для определения групп при обработке данных. В нашем случае это поле Год.

Определение групп обработки

Укажите поля для определения групп при обработке данных

Метка столбца	Имя столбца
<input type="checkbox"/> Дата (Год + Месяц)	ab COL1
<input type="checkbox"/> Количество	9.0 COL2
<input checked="" type="checkbox"/> Год	ab YEAR

☐ Переобучать модель всегда и для каждой группы
☐ Пропускать группы с ошибками
☐ Использовать кэш для результата

На следующей вкладке укажем начальный этап обработки – узел с меткой Калькулятор: Доля.

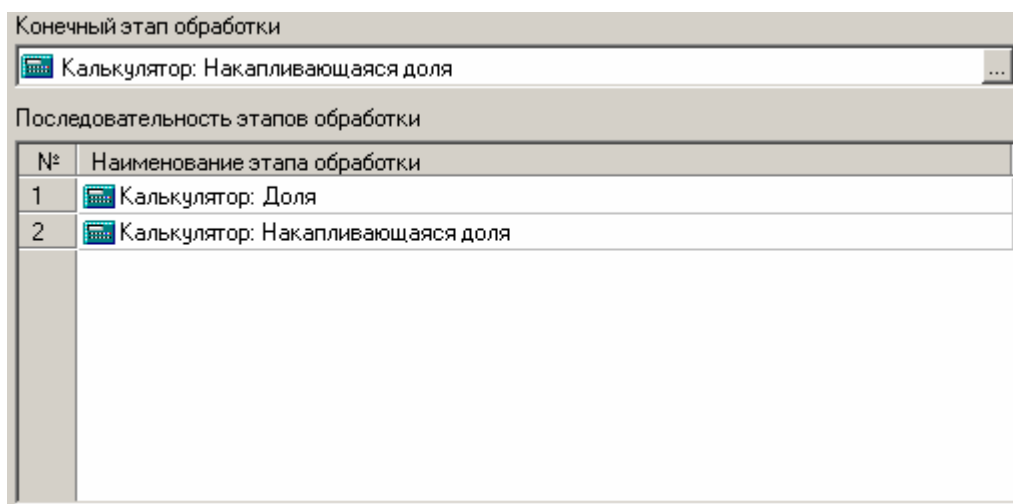
Начальный этап обработки

Калькулятор: Доля

Соответствия исходных столбцов результирующим

Исходное поле	Результирующее поле
ab Дата (Год + Месяц)	ab Дата (Год + Месяц)
9.0 Количество	9.0 Количество
ab Год	ab Год

Конечным узлом будет Калькулятор: Накапливающаяся доля.



В результате групповой обработки получим следующий набор данных (на рисунке изображен фрагмент набора).

Год	Дата (Год + Месяц)	Доля	Накопчивающаяся доля	Количество
2000	2000-M01	4,16	4	462523,419
2000	2000-M02	5,7	10	633208,196
2000	2000-M03	5,94	16	660159,299
2000	2000-M04	5,56	21	617455,3417
2000	2000-M05	5,38	27	597354,4794
2000	2000-M06	7,14	34	793517,4512
2000	2000-M07	9,15	43	1015944,2862
2000	2000-M08	10,34	53	1148052,2523
2000	2000-M09	10,41	64	1156623,1715
2000	2000-M10	11,3	75	1255021,9423
2000	2000-M11	12,7	88	1410114,5606
2000	2000-M12	12,22	100	1357230,3388
2001	2001-M01	5,16	5	1003317,7317
2001	2001-M02	5,64	11	1097048,6263
2001	2001-M03	7,71	19	1498977,3427
2001	2001-M04	7,76	26	1507696,4482
2001	2001-M05	7,82	34	1520761,5589
2001	2001-M06	8,25	42	1602674,5245
2001	2001-M07	8,67	51	1685899,1625
2001	2001-M08	9,77	61	1899255,945
2001	2001-M09	8,83	70	1716804,1633
2001	2001-M10	10,65	80	2069772,3982
2001	2001-M11	10,37	91	2016227,4267
2001	2001-M12	9,35	100	1817580,4566
2002	2002-M01	6,19	6	1493788,5092

Обратите внимание – накопчивающаяся доля доходит до 100% в каждом году, и «сбрасывается» с началом нового года. Таким образом, мы получили желаемый результат. Без групповой обработки получить это было бы гораздо сложнее.

На первой вкладке мастера настройки узла были доступны три опции. Разберем их детальнее.

Флаг **Переобучать модель всегда и для каждой группы** актуален, когда в цепочке узлов, на которые ссылается групповая обработка, имеются какие-либо модели – линейная регрессия, нейронная сеть и так далее. Поэтому в случае простых действий – **Калькулятор**,

Фильтр, Замена данных, Сортировка и другие – на данный флаг не нужно обращать внимания.

Флаг **Пропускать группы с ошибками** исключит из результирующего набора группы, при «прогоне» которых через цепочку узлов возникла ошибка. В подавляющем большинстве случаев это бывает также при наличии в цепочке узлов каких-либо моделей, поэтому при простых действиях флаг ставить не нужно.

Флаг **Использовать кэш для результата** определяет один из двух вариантов функционирования узла: «без использования кэширования» и «с использованием кэширования».

Определение

Кэш – это подборка данных, дублирующих оригинальные значения, сохранённые где-то или вычисленные ранее, когда оригинальные данные труднодоступны из-за большого времени доступа или для вычисления. Многие программы записывают куда-либо промежуточные или вспомогательные результаты работы, чтобы не вычислять их каждый раз, когда они понадобятся. Это ускоряет работу, но требует дополнительной памяти (оперативной или дисковой).

Кэш требуется для экономии памяти. Это необходимо, когда групп обработки много и каждая группа требует больших вычислительных затрат. Большие вычислительные затраты, как правило, возникают при переобучении моделей – пересчете коэффициентов регрессии, подборе весов нейронной сети и так далее. Поэтому здесь совет следующий. Когда групп немного и в цепочке узлов «прогона» групп нет моделей, то кэш не нужен. В иных случаях лучше поставить флаг с кэшем.

Практическая работа:

- 1 Повторите в Deductor пример с групповой обработкой.

Вопросы для проверки:

- 1 Для чего предназначен узел **Групповая обработка**?
- 2 В чем принципиальное отличие узла **Скрипт** от **Групповая обработка**?
- 3 Приведите примеры, когда может потребоваться **Групповая обработка**.
- 4 В каких случаях нужно включать флаг **Использовать кэш для результата**?

Занятие 9. Настройка среды Deductor Studio

Управление расположением окон

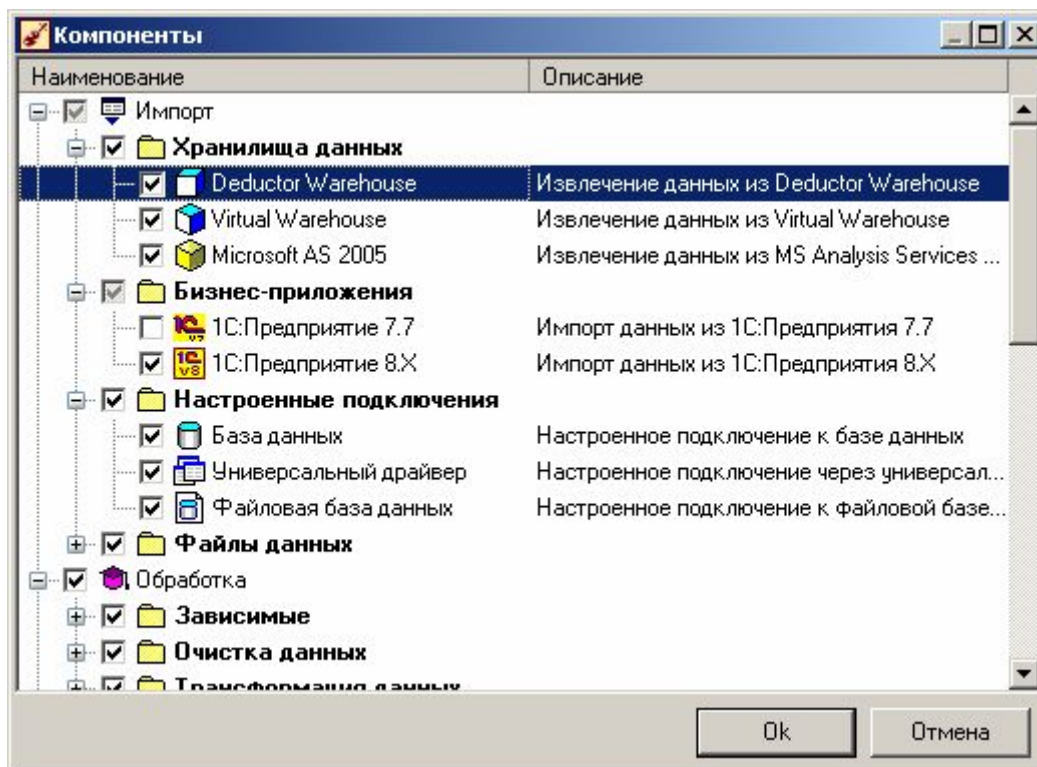
По умолчанию при открытии Deductor Studio панель управления располагается слева, а область визуализаторов – справа. Однако при помощи мыши можно отстыковать панель управления и расположить в любом месте экрана.

В меню **Окно** доступны следующие команды по отображению окон:

- Каскадом – окна располагаются одно за другим но так, что для всех окон видна строка заголовка. Щелчок по ней делает окно активным и переводит его на передний план.
- Горизонтально – в результате этой команды открытые окна с данными располагаются рядом без перекрытия по горизонтали.
- Вертикально – расположить окна с данными вертикально. Окна располагаются рядом без перекрытия по вертикали.
- Свернуть все – сворачивает все окна с данными до размеров кнопок в нижней части окна программы. Чтобы вернуть окно в прежнее состояние достаточно щелкнуть по соответствующей ему кнопке.
- Закрыть все – закрывает все окна в области визуализаторов.

Каждый открытый визуализатор узла занимает определенное количество оперативной памяти компьютера. Поэтому на компьютерах с небольшим количеством оперативной памяти не рекомендуется открывать одновременно много визуализаторов.

Доступные компоненты



В **Deductor Studio** имеется множество механизмов импорта, экспорта, обработки и визуализации, а также источников данных, но не во всех из них имеется потребность при разработке сценариев. Иногда желательно отключить видимость той или иной компоненты, причем это касается только доступности методов при вызове мастеров настройки. Т.е. если в сценарии уже используется какой-то из скрытых механизмов, то он все равно будет отображен и выполнен, но при вызове мастера для добавления нового действия в сценарий этот механизм не будет отображаться.

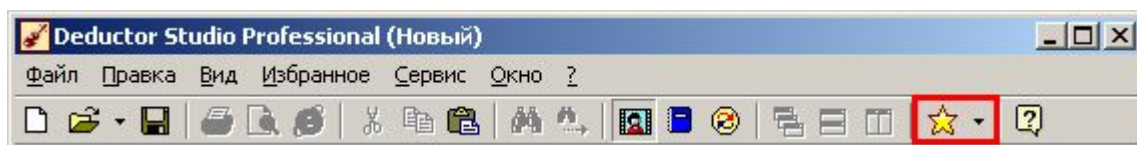
Отключая ненужные в данный момент действия можно значительно упростить работу по построению сценариев.

Настроить доступные компоненты можно в меню **Сервис ► Компоненты...** видимость можно как у отдельных компонентов, так и у целой группы.

По умолчанию все компоненты видимы.

Избранные узлы

Очень часто при создании сценариев значительная часть работы ведется с несколькими ключевыми узлами, в которых и определяются наиболее важные параметры обработки. Для упрощения работы с ними, в частности для того, чтобы их было легче найти в больших проектах в **Deductor Studio** реализована работа с избранными узлами. Избранным может быть любой сценария.




Для того чтобы добавить выделенный узел в **Избранное**:

- «Схватить» узел мышкой и перетащить его в область избранных на главной панели программы, выделенный на рисунке красным.
- Контекстное меню **Добавить в избранное** на вкладке **Сценарии**.
- Главное меню **Избранное ► Добавить в Избранное**.

При добавлении узла в **Избранное** существует возможность задать ему закладку – текстовое описание. По умолчанию закладка совпадает с меткой узла.

Для перехода к избранному узлу:

- Выбрать его из раскрывающегося списка на главной панели инструментов
- Вызвать кнопкой  окно **Избранное**, выбрать узел и нажать кнопку **Перейти**.

Файл конфигурации

В файле конфигурации хранится информация о настроенных источниках данных. По умолчанию файл конфигурации называется **Connections.sys** и находится в каталоге Мои документы\Deductor. Эти настройки можно переопределить, открыв окно командой главного меню **Сервис ► Настройка...**

Документация и демопримеры Deductor Studio

В поставке Deductor Studio, помимо справочной системы, идет комплект документации и демопримеры. Открыть список документации можно из меню Windows **Пуск ► Deductor ► Документация**. Основным документом для аналитика является **Руководство аналитика**. При решении задач консолидации и сбора информации аналитику может потребоваться **Руководство по импорту и экспорту**.

Демопримеры располагаются в директории \Samples каталога установки **Deductor**.

Практическая работа:

- 1 Откройте сценарий **Демопример по анализу данных.ded**.
- 2 Запустите на выполнение несколько узлов в различных ветках.
- 3 Расположите визуализаторы вертикально.
- 4 Сверните, а затем закройте все визуализаторы.
- 5 Добавьте несколько узлов в **Избранное**.
- 6 Отключите все незнакомые вам обработчики.
- 7 Откройте Руководство аналитика.

Вопросы для проверки:

- 1 Что делают следующие команды главного меню **Окно**: Каскадом, Вертикально, Горизонтально?
- 2 Что такое «избранные» узлы?
- 3 Как перейти к «избранному» узлу?
- 4 Как по умолчанию называется файл конфигурации?
- 5 Что хранится в файле конфигурации?
- 6 В каком каталоге находятся демопримеры?