

ГЛАВА

6

Глава 6 Работа с архивами

6.1 Введение

ПО "SyTrack-TOOL" ARC (инструмент "**Архивы**") является инструментарием, необходимым для настройки, ведения, сбора, просмотра архивов.

Сбор архивных данных в хранилище архивов может производиться:

- из стандартных архивов контроллеров Деконт (компоненты "Архив аналогов", "Архив счётчиков" и "Архив событий");
- из архивов внешних устройств, подключенных к контроллеру Деконт (например, собственные архивы теплосчетчиков, счетчиков электрической энергии);
- из архивов внешних архивов Деконта (флэш-архив);
- из хранилищ архивов на других компьютерах.

В хранилище поддерживается ведение следующих типов архивов:

- периодические аналоги;
- периодические счетчики;
- события;
- аналоги по изменению;
- счетчики по изменению;
- оперативный журнал.

Архивные данные на компьютере располагаются в **хранилище архивов**. Под хранилищем понимается база данных, управляемая сервером **InterBase** (6.0 или выше) или сервером **Firebird** (рекомендуется использовать версию 1.5 или выше), далее SQL сервер.

ПО "SyTrack-TOOL" ARC состоит из четырех основных программ (ярлыки к ним расположены в "Пуск\Программы\DeCont\Работа с архивами"):

1. [Менеджер хранилища](#)- ПО "**SyTrack-WRT**" ARC
2. [Сбор архивов](#) - ПО "**SyTrack-WRT**" ARC-ClientNet
3. [Просмотр архивов](#) -ПО "**SyTrack-WRT**" ARC-View
4. [Сервер хранилища](#) - ПО "**SyTrack-WRT**" ARC-OPCHDA



Для справки:

InterBase - это система управления реляционными базами данных, поставляемая корпорацией BORLAND для построения приложений с архитектурой "клиент-сервер".

Firebird - проект, образовавшийся, как только исходные тексты сервера InterBase стали доступными (июль 2000 года). Проект ведут бывшие сотрудники группы Interbase и сторонние разработчики. Группой исправлено большое количество ошибок и добавлена новая функциональность.

6.2 Что нового в версии 6 ?

Общие

- Работа с сервером Firebird/InterBase без использования BDE. Используется библиотека GDS32.DLL для прямого обращения к серверу. Не наблюдается нехватка ресурсов BDE.
- Авто обновление данных просмотра у клиентов по событию из записи в хранилище.
- Изменения в формате хранения данных, позволяющие увеличить время выборки данных и уменьшить объем файла хранилища.
- Добавлены новые типы архивов аналоги по изменению, счетчики по изменению и оперативный журнал.
- Справочник архивов разбит на два справочника: периодические архивы и архивы по изменению.
- Реализован набор COM объектов для доступа к данным хранилища.
- Два режима записи данных синхронный ("Сбор архивов") и асинхронный ("ОПС модель").
- Изменения в интерфейсе программ.
- Таблицы под данные создаются по факту записи данных в хранилище.
- Хранение данных в зимнем времени.
- Создание скрипта SQL-хранилища.
- Создание хранилища по SQL-скрипту.
- Суточное резервирование данных и их просмотр.

Менеджер хранилища

- Реализован одним exe-файлом в виде программы и службы.
- При старте сбора данных запускается и выгружается SQL сервером посредством вызова хранимой процедуры (при удаленном вычитывании(SQL сервер хранилища находится на другом компьютере) не требуется настраивать DCOM).
- При очистке хранилища можно накладывать фильтр по архивам.

Сбор архивов

- Новый тип вычитываемых архивов "Внешние архивы Деконта".
- Более подробный и наглядный вид окна состояния системы.
- Возможность локальной остановки вычитывания отдельного архива/архивов. Сохранение состояния запуска в хранилище.
- Вычитать не ожидая для отдельного архива/архивов.
- Выборочное вычитывание для отдельного архива/архивов.
- Установка времени запроса данных для отдельного архива/архивов.
- Обновление конфигурации вычитывания, не требующее рестарта. Связь с уже существующими серверами не разрывается.
- Запись логов работы сбора в суточный файл.
- Анализ необходимости запроса данных.

Просмотр архивов

- Авто обновление данных по факту из записи в хранилище(событийное). Данные дочитываются, а не перечитываются целиком.
- Скорость выборки данных возросла в несколько раз.
- Экспорт данных xml, html, txt, xml форматы.
- Более универсальная фильтрация данных и сортировка.
- Возможен более детальный анализ данных (в периодических архивах заметны "дыры" в данных, в архивах по изменению-неопределенные состояния параметра).
- Отображение графиков отдельно для типа архивы с разной осью Y и общей X. Графики дискретов строятся по всем значениям, а не по 0 и 1 как раньше.

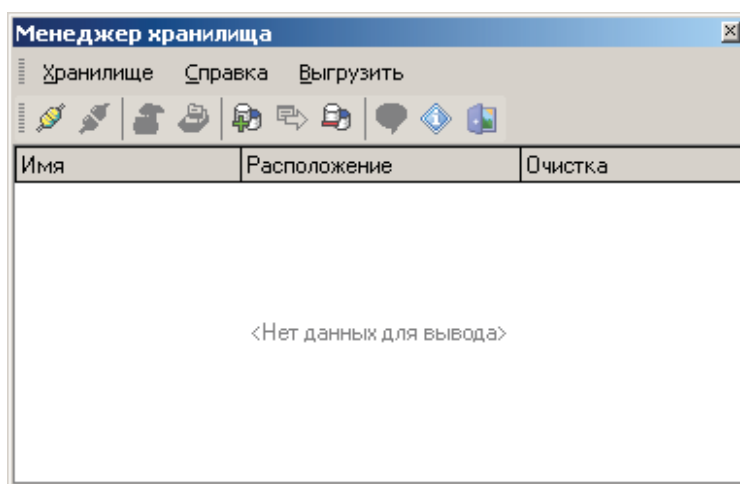
- Группа просмотра "Все дискреты" заменена на группу "Все параметры".
- Для каждого типа архива создается отдельная закладка просмотра. Три новых типа архива: аналоги по изменению и счетчики по изменению и оперативный журнал.

6.3 Менеджер хранилища

ПО "SyTrack-WRT" ARC (программа "Менеджер хранилища", далее менеджер) представляет собой инструментальное средство, предназначенное для управления хранилищами архивов, расположенных на данном компьютере. Часть функций менеджера доступна в интерактивном режиме, например, создание хранилища, просмотр и редактирование справочников. Но основное назначение менеджера - это оперативное обслуживание хранилищ, в которые ведется запись архивных данных. К такому обслуживанию относятся операции: [подчистка хранилища](#) (удаление устаревших архивных данных), резервное копирование. В каждом хранилище указывается время выполнения данных операций.

Основные функции менеджера:

- [создание хранилища](#);
- [изменение настроек хранилища](#);
- [очистка хранилища](#) (удаление архивов, для которых истёк срок хранения; очистка справочника архивов и параметров);
- [просмотр и редактирование справочников](#)
- [экспорт данных в другое хранилище](#);
- [удаление хранилища](#).



В ПО "SyTrack-WRT" ARC предусмотрено лицензирование по количеству дискретов/аналогов/счётчиков в хранилище архивов:

- до 100
- до 200
- до 300
- до 400
- от 400

6.3.1 Создание хранилища

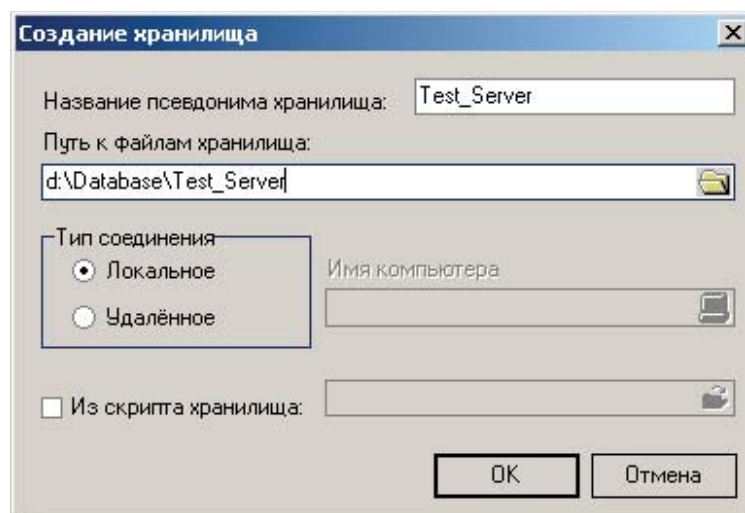
Для создания **нового хранилища** выберите пункт меню "Хранилище -> Создать новое..." или нажмите на соответствующую кнопку на главном окне программы.

В появившемся окне нужно указать:

- Название псевдонима хранилища (любое имя, которое впоследствии будет использоваться для соединения с хранилищем);
- Путь к файлам хранилища (каталог, в котором будут расположены файлы архивного хранилища);
- Для создания удаленного хранилища (хранилище, расположенное на другом компьютере) требуется установить тип соединения "Удаленное", а также в поле "Имя компьютера" имя удаленного компьютера в локальной сети. При этом будет производиться взаимодействие с SQL сервером по протоколу TCP/IP, запущенным, на указанном компьютере, а не с локальным.
- Если требуется создать хранилище по скрипту, то укажите что хранилище будет создано из скрипта и задайте имя файла скрипта хранилища. (см. также [Создание скрипта хранилища](#)).

Возможные ошибки при создании хранилища (при указанных ситуациях хранилище не будет создано):

- указана папка архивного хранилища, которая не существует и которую нельзя создать;
- в указанную папку нет доступа для записи (при создании удаленного хранилища не требуется давать сетевой доступ к папке на удаленном компьютере, достаточно, чтобы процесс SQL сервера имел локальные права для записи в соответствующую папку);
- в указанной папке уже существует хранилище, с которым сервер поддерживает соединение.



6.3.2 Создание скрипта хранилища

Скрипт хранилища - это текстовый файл, состоящий из SQL-команд, создающих точную копию структуры хранилища с заданными параметрами, архивами, группами просмотра, конфигурацией сбора данных и т.д. за исключением данных. Для создания скрипта хранилища выберите пункт меню "Хранилище -> Создать скрипт хранилища...", указав далее расположение и название файла скрипта хранилища. Для создания хранилища по заданному скрипту см. [Создание хранилища](#).

6.3.3 Соединение с хранилищем и его обслуживание

Любая программа перед записью архивных данных устанавливает соединение с хранилищем (например, программа "[Сбор архивов](#)"). В этот момент менеджер автоматически загружается и устанавливает соединение с нужным хранилищем, на том компьютере, где оно находится. При этом менеджер "активируется", т.е. в определённое время [очищает хранилище](#) и создает резервные копии. Менеджер способен одновременно обслуживать любое количество хранилищ. Информация об обслуживаемых в данное время хранилищах выдвается в главном окне программы в виде двухуровневого дерева. Узлами дерева являются названия обслуживаемых хранилищ, а листьями - информация о том, с какого компьютера происходит вычитывание архивов и название псевдонима хранилища на этом компьютере.

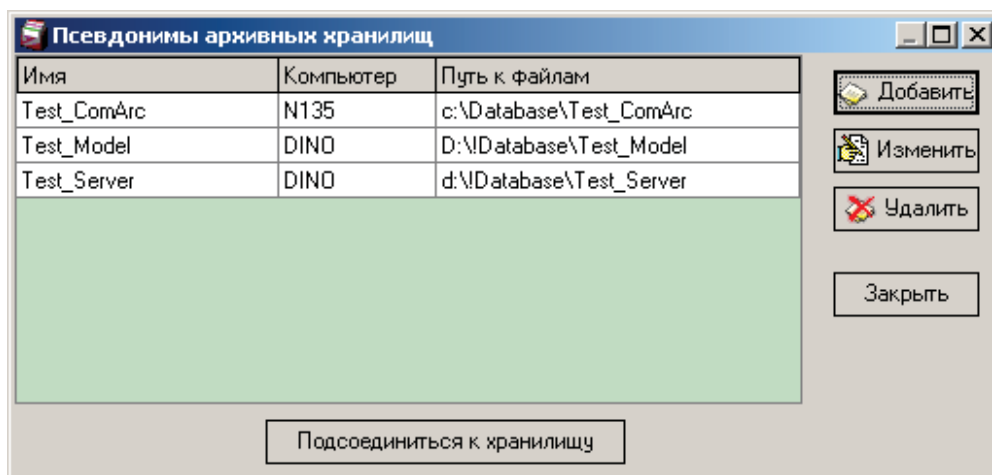
Таким образом, менеджер способен обслуживать одновременно несколько хранилищ и давать возможность писать данные нескольким клиентам в одно и то же хранилище.

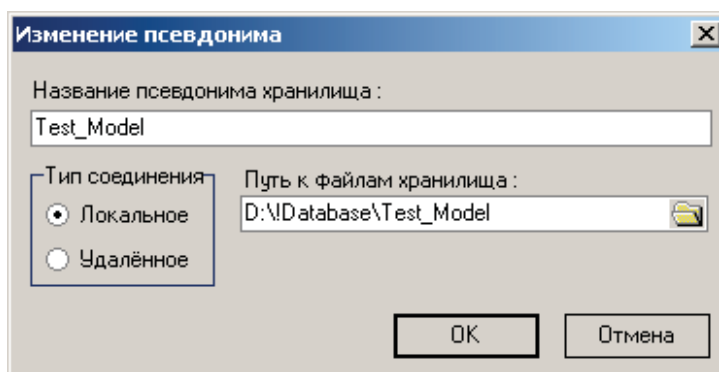
После выбора в дереве обслуживаемого хранилища доступны следующие действия и информация об хранилище (выбор главного меню программы):

- [Создание пропущенных архивов](#);
- [Очистка хранилища](#);
- [Справочники хранилища](#);
- [Настройки хранилища](#);

Имеется также возможность ручного подсоединения одного хранилища. Для этого выберите пункт меню "Хранилище -> Подсоединить хранилище...". Появится стандартное окно с выбором псевдонима хранилища. В данном окне можно как изменить выбранный псевдоним, так и добавить новый. Настройки псевдонима аналогичны настройкам при [создании нового хранилища](#).

При этом хранилище будет принято на обслуживание менеджером, как было описано выше.





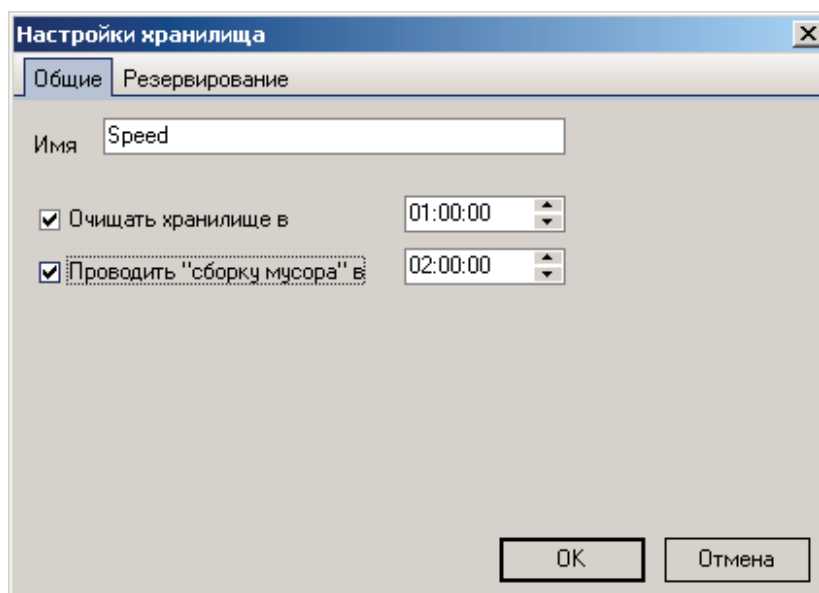
6.3.4 Настройки хранилища

Окно настроек хранилища можно вызвать, выбрав пункт меню "Хранилище ->Конфигурация".

В появившемся окне можно установить:

- имя хранилища(хранится в самом хранилище и по умолчанию соответствует псевдониму, указанному при создании хранилища);
- очищать или нет хранилище и в какое время раз в сутки производить его очистку (см. [Очистка хранилища](#));
- проводить или нет сборку мусора и в какое время раз в сутки. "Сборка мусора подразумевает удаление старых версий записей в базе данных, что связано с особенностью работы версионных серверов база данных, к которым относится Interbase/Firebird.

Для настройки резервирования данных в хранилище см. [Резервное копирование данных](#).



6.3.5 Резервное копирование данных

Резервное копирование данных хранилища представляет из себя копирование данных из основного хранилища в резервное хранилище. Резервное хранилище может находиться в локальной папке, а также удаленно на другом компьютере. Резервное хранилище разбито в виде файлов по суткам. Формат резервных файлов: YY-MM-DD.gdb(zip), где YY-последние две цифры года, MM-номер месяца в году, DD-день месяца. Возможно проводить упаковку резервных файлов с глубиной максимальной актуальности данных для архивов. Резервное копирование данных производится только за предыдущие дни, за текущий день не производится.

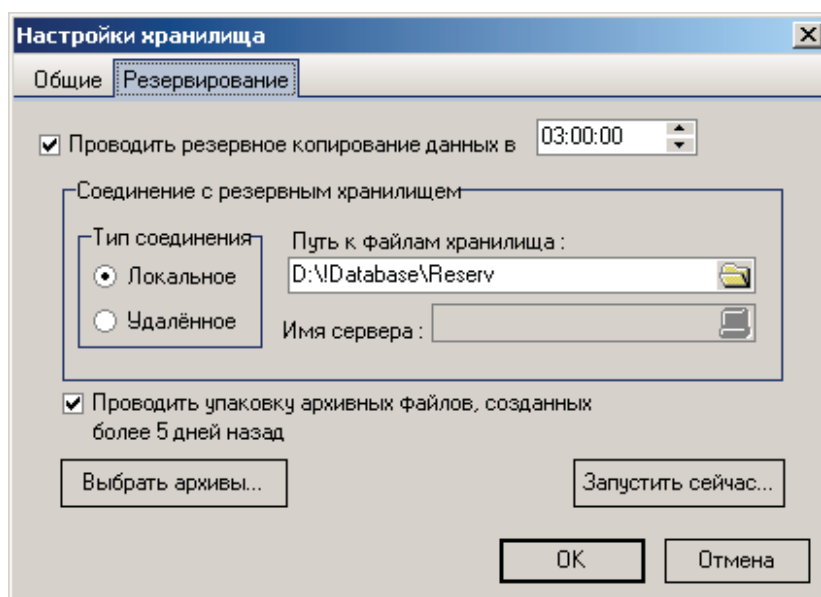
Для проведения резервного копирования данных, требуется провести его настройку. Для этого выберите пункт меню "Хранилище ->Конфигурация" и перейдите на закладку "Резервирование". В появившемся окне (см. рис) требуется указать:

- Необходимо ли проводить резервное копирование и в какое время суток это делать;
- Папку для резервных файлов хранилища. Если указана удаленная папка (Тип соединения - удаленное и задано Имя сервера), то требуется, чтобы на удаленном компьютере был установлен Interbase/Firebird сервер и программное обеспечение "Работа с архивами";
- Необходимо ли проводить упаковку файлов, которые датируются раньше, чем максимальная актуальность данных для выбранных архивов;
- Архивы, подлежащие резервному копированию (кнопка "Выбрать архивы...").

После настройки резервного копирования его можно запустить в ручном режиме (кнопка "Запустить сейчас...").



По умолчанию резервное копирование не производится.



6.3.6 Создание архивов

Записываемые данные в хранилище хранятся в таблицах, которые создаются и удаляются динамически в соответствии с настройками хранилища. Данные группируются по таблицам в соответствии с их меткой времени, типом данных и их периодом (если данные являются периодическими аналогами или счётчиками). Это обеспечивает структурированность данных, более быстрый к ним доступ. Таблицы создаются в хранилище по мере их записи данных.



Имена таблиц в хранилище формируются следующим образом:

- «**Тип архива**»
- «**Период архива**»
- «**Тип периода**»
- «**Дата архива**».

«Тип архива» в данной версии программы может принимать три значения: ANALOG, COUNT, EVENT, что соответственно означает "Архив аналогов", "Архив счётчиков" и "Архив дискретов".

«Тип периода» может принимать значения "S", "D", "M", "Q" и "Y", что соответствует секундам, суткам, месяцам, кварталам и годам.

Если тип периода – секунды, то «Период архива» записывается в формате "HH_MM_SS" (для EVENT - отсутствует). Например, "00_00_30" соответствует полуминутным данным, 01_00_00 – часовым данным.

Если тип периода принимает другие значения, то он записывается в формате "NN", где NN – количество периодов соответствующего типа.

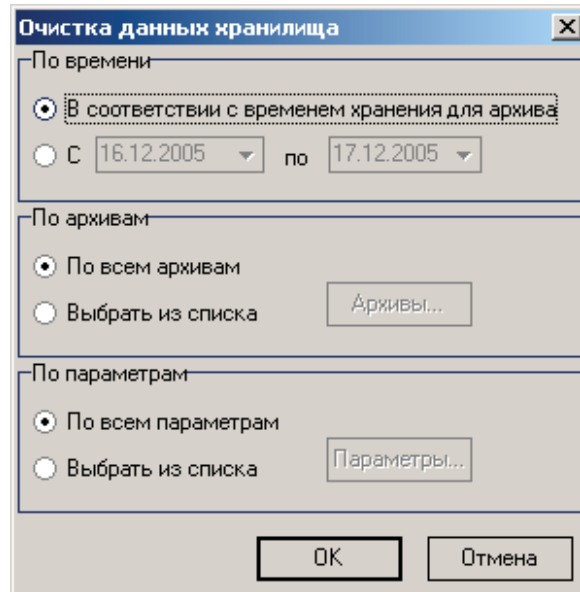
«Дата архива» - в формате "DD_MM_YY". Например, 03_05_01 – архив за 3 мая 2001 года.

6.3.7 Очистка хранилища

Очистка хранилища подразумевает под собой удаление "старых", ненужных данных из хранилища, что позволяет экономить место на жестком диске, а также уменьшить вычислительные затраты при выборке данных. Очистку хранилища рекомендуется производить во время наименьшей загрузки системы, т.к. она может занимать долгое время и в какой-то мере замедлить другие процессы в системе.

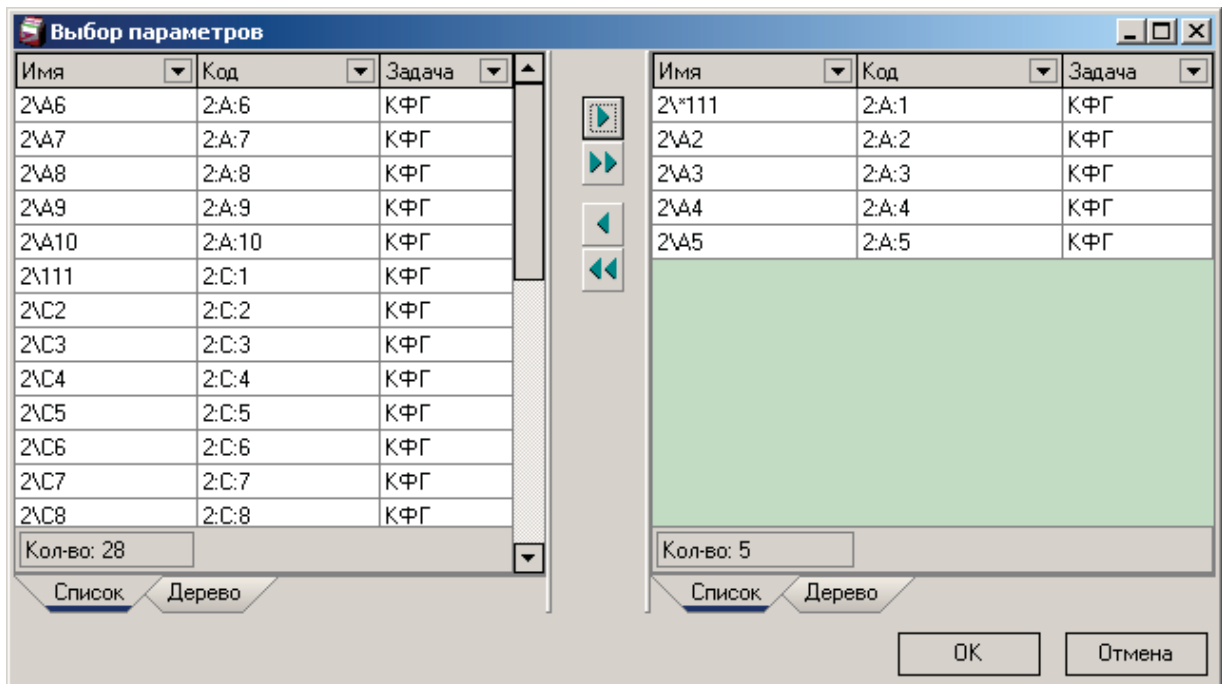
Очистку хранилища можно производить:

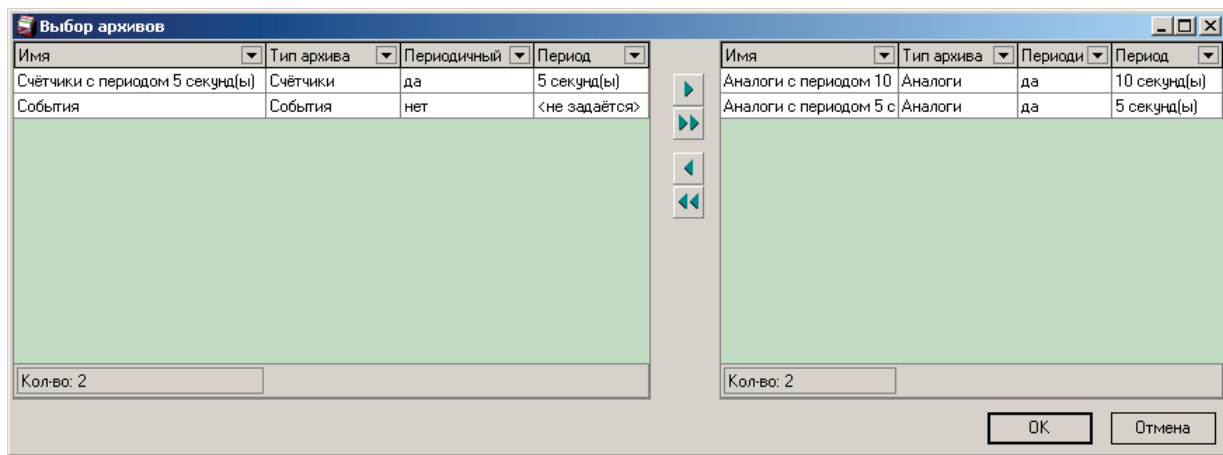
1. Автоматически в заданное время. Время очистки указывается [в настройках хранилища](#).
2. По требованию, выбрав пункт меню "Хранилище ->Очистка хранилища" в главном окне программы. При этом появится окно с настройками очистки хранилища.



По умолчанию в данном окне установлены настройки, используемые при автоматической очистке. При этом длительность хранения данных рассчитывается в соответствии с "**Периодом хранения**" из [справочника архивов](#), и очистка производится по всем параметрам хранилища.

При очистке по требованию можно выбрать временной интервал очистки хранилища, а также указать параметры и архивы, данные из которых будут подвержены очистке. Выбор параметров производится в окне, появляющемся при нажатии кнопки <Параметры...>, а выбор архивов по кнопке <Архивы...>. Параметры могут быть представлены в виде списка или в виде дерева выбором соответствующей закладки. Из левого списка требуемые параметры нужно переместить в правый список.





Очистку хранилища условно можно разделить на следующие этапы:

- **Очистка архивов:**

В процессе данного этапа удаляются все архивы (на уровне баз данных - таблицы), для которых истёк срок хранения (в днях), указанный в [справочнике архивов](#).

Архивы, для которых установлен срок хранения 0 дней, хранятся неограниченное количество времени.

- **Очистка справочника параметров:**

Справочник параметров очищается по следующему правилу для каждого параметра:

из [справочника параметров](#) удаляется запись о параметре, для которого флаг "Удалять" установлен в значение "да", и нет ни в одном архиве связанных с ним данных.

- **Очистка справочника архивов:**

Из [справочника архивов](#) удаляются те записи, для которых флаг "Удалять" установлен в "да", и нет ни одного архива (таблицы), описанного данной записью. Тем самым помеченный на удаление архив удаляется не сразу, а по истечении своего срока хранения. Для того, чтобы удалить весь архив быстрее, достаточно уменьшить его срок хранения до одного дня, и при следующей очистке он будет удалён.

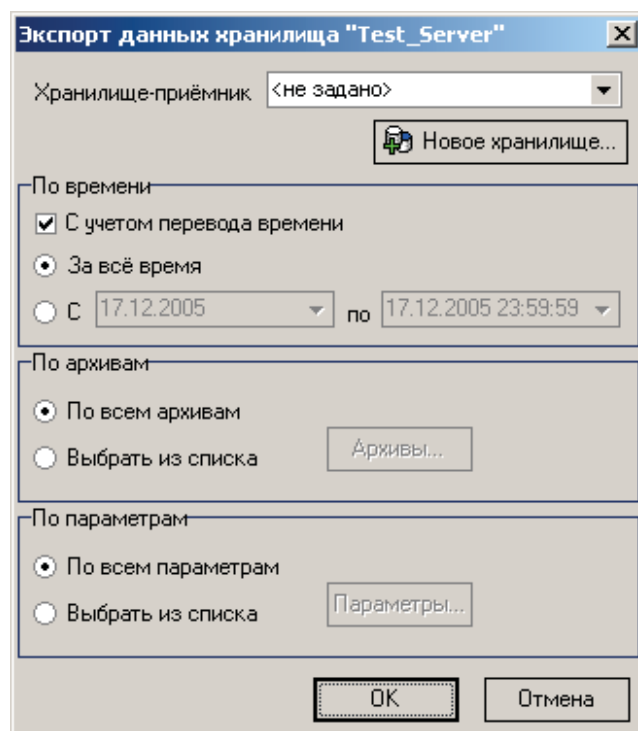
6.3.8 Экспорт данных в другое хранилище

Окно настроек экспорта данных в другое хранилище можно вызвать, выбрав пункт меню "Хранилище ->Экспорт в другое хранилище...".

Процедура экспорта данных позволяет произвести копирование данных из одного хранилища в другое по выбранным архивам и параметрам, а также по временной метке для данных.

Выбор параметров осуществляется аналогично выбору параметров при [очистке хранилища](#).

Хранилище-приёмник может быть новым и создано нажатием кнопки <Новое хранилище...> (см. [Создание хранилища](#)).



6.3.9 Репликация данных

ПО "SyTrack-WRT" ARC-HOTRES (компонент "Резервирование архивов на ПК") позволяет производить резервное копирование данных из основного хранилища в резервное. Резервное хранилище может находиться в локальной папке, а также удаленно на другом компьютере.

Также компонент "Резервирование архивов на ПК" помогает организовать отказоустойчивый кластер.



Для организации кластера необходимо иметь 2 компьютера. На обоих компьютерах создаются архивы, которые ведутся [моделью](#) в программе [WinDecont](#). Чтобы архивы велись, необходимо в настройках [модели](#) в программе [WinDecont](#) указать архив, в который нужно архивировать данные. Также необходимо настроить репликацию данных между архивами на обоих компьютерах. Контролем состояния и переключением серверов кластера занимается программа "[Резервирование для ПК: организация отказоустойчивого кластера для дублирования компьютеров](#)".

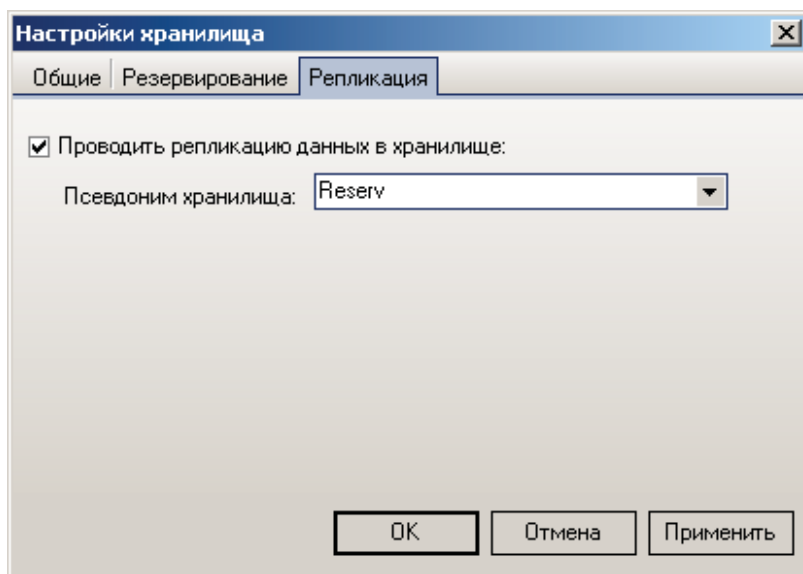
Под репликацией данных подразумевается механизм синхронизации данных с другим хранилищем. Данный механизм позволяет повысить надёжность системы хранения данных и обеспечить быстрый доступ к ним. Репликацию данных целесообразно проводить на другой компьютер в локальной сети. Перед этим потребуется создать хранилище, с которым будет производиться синхронизация данных.

Для проведения репликации данных, требуется провести её настройку. Для этого выберите пункт меню "Хранилище - >Конфигурация" и перейдите на закладку "Репликация". В появившемся окне (см. рис) требуется указать:

- Нужно или нет проводить репликацию (по умолчанию репликация не производится);
- Псевдоним хранилища, в которое будет производиться репликация данных.

При репликации следует учитывать следующие моменты:

- Репликация данных производится с момента её настройки, т.е. данные записанные в хранилище до настройки репликации не будут синхронизированы с другим хранилищем;
- Псевдоним хранилища, в которое будет производиться репликация данных.
- Если хранилище, с которым производится синхронизация данных, становится недоступно, то данные кэшируются на записываемой стороне в подпапке "sync" папки расположения хранилища. Синхронизация данных возобновится по мере доступности хранилища, с которым производится синхронизация. Кэшируемые данные удаляются по мере синхронизации с другим хранилищем.



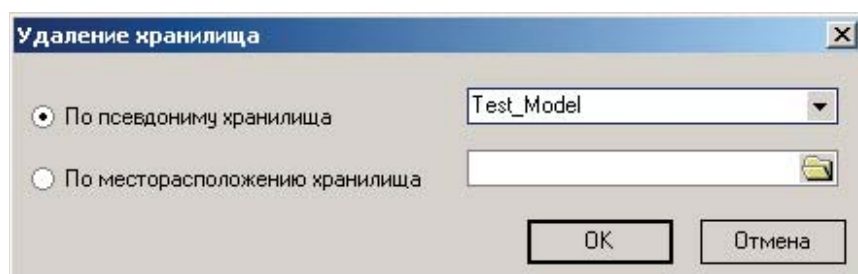
6.3.10 Удаление хранилища

Окно удаления хранилища можно вызвать, выбрав пункт меню "Хранилище ->Удалить хранилище..." или нажав соответствующую кнопку на главном окне программы.

В появившемся окне "**Удаление хранилища**" можно выбрать один из двух режимов удаления хранилища:

1. По псевдониму хранилища. В выпадающем списке нужно выбрать псевдоним (см. [Соединение с хранилищем и его обслуживание](#)), указывающий на удаляемое хранилище. При этом режиме псевдоним удаляется вместе с хранилищем.
2. По месторасположению хранилища. В соответствующем поле указывается папка (путь к архивному хранилищу), которое будет удаляться.

Перед удалением архивного хранилища, с которым менеджер в текущее время поддерживает соединение, нужно от него отсоединиться. Также не должно быть других клиентов подсоединенных к удаляемому хранилищу.



6.3.11 Справочники хранилища

Для просмотра справочников хранилища выберете пункт меню "Хранилище->Справочники" или нажмите соответствующую кнопку на главном окне программы.

Для выбора нужного справочника воспользуйтесь панелью слева. Справочники хранилища разбиты на категории по их назначению.

Доступны следующие справочники:

- [Справочник параметров](#)
- [Справочник периодических архивов](#)
- [Справочник архивов по изменению](#)
- Справочник настройки вычитывания архивов Деконта
- Справочник настройки вычитывания архивов внешних устройств
- Справочник настройки вычитывания внешних архивов Деконта
- Справочник настройки вычитывания архивов удаленного хранилища

Имя	Код	Уда- лять	Зад- ача	Расшифровка состояния
1 204\D1	204:D:1	<input checked="" type="checkbox"/>	КФГ	Включено/Отключено
2 204\D2	204:D:2	<input type="checkbox"/>	КФГ	Включено/Отключено
3 204\D3	204:D:3	<input type="checkbox"/>	КФГ	Включено/Отключено
4 204\D4	204:D:4	<input type="checkbox"/>	КФГ	Включено/Отключено
5 204\D10	204:D:10	<input type="checkbox"/>	КФГ	Включено/Отключено
6 204\D11	204:D:11	<input type="checkbox"/>	КФГ	Включено/Отключено
7 204\D12	204:D:12	<input type="checkbox"/>	КФГ	Включено/Отключено
8 204\D50	204:D:50	<input type="checkbox"/>	КФГ	Включено/Отключено
9 204\D51	204:D:51	<input type="checkbox"/>	КФГ	Включено/Отключено
10 204\D52	204:D:52	<input type="checkbox"/>	КФГ	Включено/Отключено
11 204\D53	204:D:53	<input type="checkbox"/>	КФГ	Включено/Отключено
12 204\D54	204:D:54	<input type="checkbox"/>	КФГ	Включено/Отключено
13 204\D55	204:D:55	<input type="checkbox"/>	КФГ	Включено/Отключено
14 204\D56	204:D:56	<input type="checkbox"/>	КФГ	Включено/Отключено
15 204\D57	204:D:57	<input type="checkbox"/>	КФГ	Включено/Отключено
16 204\D58	204:D:58	<input type="checkbox"/>	КФГ	Включено/Отключено
17 204\D59	204:D:59	<input type="checkbox"/>	КФГ	Включено/Отключено
18 204\D60	204:D:60	<input type="checkbox"/>	КФГ	Включено/Отключено
19 204\D61	204:D:61	<input type="checkbox"/>	КФГ	Включено/Отключено
20 204\D62	204:D:62	<input type="checkbox"/>	КФГ	Включено/Отключено
21 204\D63	204:D:63	<input type="checkbox"/>	КФГ	Включено/Отключено
22 204\D64	204:D:64	<input type="checkbox"/>	КФГ	Включено/Отключено
23 204\D65	204:D:65	<input type="checkbox"/>	КФГ	Включено/Отключено

6.3.11.1 Справочник параметров

Справочник параметров представляет собой таблицу, где каждая строка описывает отдельный параметр.

Для каждого параметра определены следующие поля:

- **ID** – уникальный идентификатор параметра в пределах хранилища;
- **Имя параметра** – любое произвольное имя, уникальное внутри задачи (не более 127 символов);
- **Код параметра** – уникальный код параметра внутри задачи (не более 127 символов);
- **Код задачи** – служит признаком для группировки параметров для отдельной задачи (не более 127 символов);
- **Таблица расшифровки** – служит для текстового пояснения состояния события (поле действительно только для архива событий);
- **Удалять** – флаг удаления. Если параметр не используется, и данное поле имеет значение "да", то описание параметра удаляется при очистке хранилища.

ID	Имя	Код	Удалять	Задача	Расшифровка состояния
740	202\У7\Накопленная энергия активная обратная по тарифу 5 за данный период	202:D7:A:26	<input type="checkbox"/>	КФГ	Нет таблицы для расшифровки состояния
741	202\У7\Накопленная энергия реактивная прямая по тарифу 5 за данный период	202:D7:A:27	<input type="checkbox"/>	КФГ	Нет таблицы для расшифровки состояния
742	202\У7\Накопленная энергия реактивная обратная по тарифу 5 за данный период	202:D7:A:28	<input type="checkbox"/>	КФГ	Нет таблицы для расшифровки состояния
743	202\У7\Накопленная энергия активная прямая по тарифу 6 за данный период	202:D7:A:29	<input type="checkbox"/>	КФГ	Нет таблицы для расшифровки состояния
744	202\У7\Накопленная энергия активная обратная по тарифу 6 за данный период	202:D7:A:30	<input type="checkbox"/>	КФГ	Нет таблицы для расшифровки состояния
745	202\У7\Накопленная энергия реактивная прямая по тарифу 6 за данный период	202:D7:A:31	<input type="checkbox"/>	КФГ	Нет таблицы для расшифровки состояния
746	202\У7\Накопленная энергия реактивная обратная по тарифу 6 за данный период	202:D7:A:32	<input type="checkbox"/>	КФГ	Нет таблицы для расшифровки состояния
747	202\У7\Накопленная энергия активная прямая по тарифу 7 за данный период	202:D7:A:33	<input type="checkbox"/>	КФГ	Нет таблицы для расшифровки состояния
748	202\У7\Накопленная энергия активная обратная по тарифу 7 за данный период	202:D7:A:34	<input type="checkbox"/>	КФГ	Нет таблицы для расшифровки состояния
749	202\У7\Накопленная энергия реактивная прямая по тарифу 7 за данный период	202:D7:A:35	<input type="checkbox"/>	КФГ	Нет таблицы для расшифровки состояния
750	202\У7\Накопленная энергия реактивная обратная по тарифу 7 за данный период	202:D7:A:36	<input type="checkbox"/>	КФГ	Нет таблицы для расшифровки состояния
751	202\У7\Накопленная энергия активная прямая по тарифу 8 за данный период	202:D7:A:37	<input type="checkbox"/>	КФГ	Нет таблицы для расшифровки состояния
752	202\У7\Накопленная энергия активная обратная по тарифу 8 за данный период	202:D7:A:38	<input type="checkbox"/>	КФГ	Нет таблицы для расшифровки состояния
753	202\У7\Накопленная энергия реактивная прямая по тарифу 8 за данный период	202:D7:A:39	<input type="checkbox"/>	КФГ	Нет таблицы для расшифровки состояния
754	202\У7\Накопленная энергия реактивная обратная по тарифу 8 за данный период	202:D7:A:40	<input type="checkbox"/>	КФГ	Нет таблицы для расшифровки состояния
755	202\У1\Включение/Выключение электросчетчика	202:D1:D:1	<input type="checkbox"/>	КФГ	Включено/Отключено
756	202\У2\Включение/Выключение электросчетчика	202:D2:D:1	<input type="checkbox"/>	КФГ	Включено/Отключено
757	202\У3\Включение/Выключение электросчетчика	202:D3:D:1	<input type="checkbox"/>	КФГ	Включено/Отключено
758	202\У4\Включение/Выключение электросчетчика	202:D4:D:1	<input type="checkbox"/>	КФГ	Включено/Отключено
759	202\У5\Включение/Выключение электросчетчика	202:D5:D:1	<input type="checkbox"/>	КФГ	Включено/Отключено
760	202\У6\Включение/Выключение электросчетчика	202:D6:D:1	<input type="checkbox"/>	КФГ	Включено/Отключено
761	202\У7\Включение/Выключение электросчетчика	202:D7:D:1	<input type="checkbox"/>	КФГ	Включено/Отключено
762	203\D1	203:D:1	<input type="checkbox"/>	КФГ	Включено/Отключено
763	203\D2	203:D:2	<input type="checkbox"/>	КФГ	Включено/Отключено
764	203\D3	203:D:3	<input type="checkbox"/>	КФГ	Включено/Отключено
765	203\D4	203:D:4	<input type="checkbox"/>	КФГ	Включено/Отключено
766	203\D10	203:D:10	<input type="checkbox"/>	КФГ	Включено/Отключено
767	203\D11	203:D:11	<input type="checkbox"/>	КФГ	Включено/Отключено
768	203\D12	203:D:12	<input type="checkbox"/>	КФГ	Включено/Отключено
769	203\A41_тек	203:A:41_тек	<input type="checkbox"/>	КФГ	Нет таблицы для расшифровки состояния
770	203\C1	203:C:1	<input type="checkbox"/>	КФГ	Нет таблицы для расшифровки состояния
771	203\C2	203:C:2	<input type="checkbox"/>	КФГ	Нет таблицы для расшифровки состояния

6.3.11.2 Справочник периодических архивов

Справочник архивов представляет собой таблицу. Каждая строка представляет собой описание отдельного архива.

Для каждого архива определены следующие поля:

- **ID** – уникальный идентификатор архива пределах хранилища;
- **Название** – любое произвольное имя архива (не более 100 символов);
- **Тип архива** – может принимать только одно из трёх значений, выбираемых из выпадающего списка:
"Аналоги" – архив аналогов;
"Счётчики" – архив счётчиков;
"События" – архив событий.
- **Тип периода и период** и образуют временной интервал усреднения данных для аналогов, для счётчиков - период записи его текущего значения;
- **Актуальность данных (в днях)** - указывает глубину, с которой могут быть записаны данные в хранилище исключая текущий день. Данные также будут записаны за текущий или следующий день. Если данный параметр равен 0, то данные могут быть записаны с любой датой.
- **Период хранения** – указывает время хранения архива в днях (см. [Очистка архивов](#)). Для бессрочного хранения архива нужно ввести 0);
- **Удалять** – флаг удаления. Если архивов с данным описанием не найдено и данное поле имеет значение "да", то описание архива удаляется при очистке хранилища.
- **Формат поля "Дата и время"** задает строку формата, в соответствии с которой программа "[Просмотр архивов](#)" выводит дату данных.

ID	Имя	Тип архива	Тип периода	Период	Актуальность данных (в днях)	Период хранения(день)	Удалять	Формат поля "Дата и время"
1	Счётчики с периодом 1 сутки(ок)	Счётчики	Сутки	1	5	0	<input checked="" type="checkbox"/>	dd.mm.yy
2	Аналоги с периодом 30 минута(ы)	Аналоги	Минута	30	5	0	<input type="checkbox"/>	dd.mm.yy hh:mm
3	Аналоги с периодом 1 сутки(ок)	Аналоги	Сутки	1	5	0	<input type="checkbox"/>	dd.mm.yy
4	Аналоги с периодом 1 месяц(а)	Аналоги	Месяц	1	155	0	<input type="checkbox"/>	mmmm yyyy г.
5	Аналоги с периодом 1 год(а)	Аналоги	Год	1	1825	0	<input type="checkbox"/>	yyyy г.
6	Счётчики с периодом 1 секунд(ы)	Счётчики	Секунда	1	5	0	<input type="checkbox"/>	dd.mm.yy hh:mm:ss
7	Аналоги с периодом 5 минута(ы)	Аналоги	Минута	5	5	0	<input type="checkbox"/>	dd.mm.yy hh:mm
8	Счётчики с периодом 1 месяц(а)	Счётчики	Месяц	1	155	0	<input type="checkbox"/>	mmmm yyyy г.
9	Счётчики с периодом 30 минута(ы)	Счётчики	Минута	30	5	0	<input type="checkbox"/>	dd.mm.yy hh:mm

6.3.11.3 Справочник архивов по изменению

Справочник архивов представляет собой таблицу. Каждая строка представляет собой описание отдельного архива.

Для каждого архива определены следующие поля:

- **ID** – уникальный идентификатор архива пределах хранилища;
- **Название** – любое произвольное имя архива (не более 100 символов);
- **Тип архива** – может принимать только одно из четырех значений, выбираемых из выпадающего списка:
"Аналоги по изменению" – архив аналогов;
"Счетчики по изменению" – архив счетчиков;
"События" – архив событий;
"Оперативный журнал";
- **Актуальность данных (в днях)** - указывает глубину, с которой могут быть записаны данные в хранилище исключая текущий день. Данные также будут записаны за текущий или следующий день. Если данный параметр равен 0, то данные могут быть записаны с любой датой.
- **Период хранения** – указывает время хранения архива в днях (см. [Очистка архивов](#)). Для бессрочного хранения архива нужно ввести 0);
- **Удалять** – флаг удаления. Если архивов с данным описанием не найдено и данное поле имеет значение "да", то описание архива удаляется при очистке хранилища.
- **Формат поля "Дата и время"** задает строку формата, в соответствии с которой программа "[Просмотр архивов](#)" выводит дату данных.

ID	Имя	Тип архива	Период хранения(день)	Удалять	Актуальность данных (в днях)	Формат поля "Дата и время"
1	События	События	0	<input checked="" type="checkbox"/>		5 dd.mm.yy hh:mm:ss.zzz
2	Аналоги по изменению	Аналоги по изменению	0	<input type="checkbox"/>		5 dd.mm.yy hh:mm:ss.zzz
3	Счетчики по изменению	Счётчики по изменени	0	<input type="checkbox"/>		5 dd.mm.yy hh:mm:ss.zzz

6.3.11.4 Справочник расшифровки состояний дискретов

Справочник состояний событий служит для расшифровки их состояний при просмотре данных программой "[Просмотр архивов](#)". Для каждого события можно указать свою таблицу расшифровки состояний (см. [Справочник параметров](#)).

Для создания новой таблицы расшифровки нужно добавить новую строку в таблицу задать имя таблицы расшифровки и определить для нее набор состояний из значения события и соответствующей строки расшифровки.

ID	Имя																														
1	Включено/Отключено																														
<table border="1"> <thead> <tr> <th>ID</th> <th>Имя состояния</th> <th>Значение события</th> </tr> </thead> <tbody> <tr> <td>13</td> <td>Отключено</td> <td>0</td> </tr> <tr> <td>14</td> <td>Включено</td> <td>1</td> </tr> </tbody> </table>		ID	Имя состояния	Значение события	13	Отключено	0	14	Включено	1																					
ID	Имя состояния	Значение события																													
13	Отключено	0																													
14	Включено	1																													
2	Запуск контроллера																														
<table border="1"> <thead> <tr> <th>ID</th> <th>Имя состояния</th> <th>Значение события</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Запуск контроллера</td> <td>1</td> </tr> </tbody> </table>		ID	Имя состояния	Значение события	1	Запуск контроллера	1																								
ID	Имя состояния	Значение события																													
1	Запуск контроллера	1																													
4	Перевод времени в контроллере																														
<table border="1"> <thead> <tr> <th>ID</th> <th>Имя состояния</th> <th>Значение события</th> </tr> </thead> <tbody> <tr> <td>12</td> <td>Конец перевода времени</td> <td>0</td> </tr> </tbody> </table>		ID	Имя состояния	Значение события	12	Конец перевода времени	0																								
ID	Имя состояния	Значение события																													
12	Конец перевода времени	0																													
3	Останов контроллера																														
<table border="1"> <thead> <tr> <th>ID</th> <th>Имя состояния</th> <th>Значение события</th> </tr> </thead> <tbody> <tr> <td>2</td> <td>Отключение питания</td> <td>0</td> </tr> <tr> <td>3</td> <td>Рестарт из отладчика</td> <td>1</td> </tr> <tr> <td>4</td> <td>Рестарт по кнопке "Сброс"</td> <td>2</td> </tr> <tr> <td>5</td> <td>Рестарт по срабатыванию охранного таймера</td> <td>3</td> </tr> <tr> <td>6</td> <td>Рестарт после переливки</td> <td>4</td> </tr> <tr> <td>7</td> <td>Рестарт по срабатыванию защиты ядра</td> <td>5</td> </tr> <tr> <td>8</td> <td>Рестарт по запросу верхнего уровня</td> <td>16</td> </tr> <tr> <td>9</td> <td>Рестарт в результате ошибки программного компонента</td> <td>17</td> </tr> <tr> <td>10</td> <td>Рестарт по запросу канального уровня</td> <td>18</td> </tr> </tbody> </table>		ID	Имя состояния	Значение события	2	Отключение питания	0	3	Рестарт из отладчика	1	4	Рестарт по кнопке "Сброс"	2	5	Рестарт по срабатыванию охранного таймера	3	6	Рестарт после переливки	4	7	Рестарт по срабатыванию защиты ядра	5	8	Рестарт по запросу верхнего уровня	16	9	Рестарт в результате ошибки программного компонента	17	10	Рестарт по запросу канального уровня	18
ID	Имя состояния	Значение события																													
2	Отключение питания	0																													
3	Рестарт из отладчика	1																													
4	Рестарт по кнопке "Сброс"	2																													
5	Рестарт по срабатыванию охранного таймера	3																													
6	Рестарт после переливки	4																													
7	Рестарт по срабатыванию защиты ядра	5																													
8	Рестарт по запросу верхнего уровня	16																													
9	Рестарт в результате ошибки программного компонента	17																													
10	Рестарт по запросу канального уровня	18																													

6.4 Сбор архивов

ПО "SyTrack-WRT" ARC-ClientNet (программа "Сбор архивных данных из контроллеров технологической сети") предназначено для вычитывания архивных данных в хранилище архивов на компьютере.

Вычитывание архивных данных может производиться:

- из стандартных архивов контроллеров Деконт (компоненты "Архив аналогов", "Архив счётчиков" и "Архив событий");
- из архивов внешних устройств, подключенных к контроллеру Деконт (например, собственные архивы теплосчетчиков, счетчиков электрической энергии);
- из архивов внешних архивов Деконта (флэш-архив);
- из хранилищ архивов на других компьютерах.

Перед запуском программы требуется задать **конфигурацию вычитываемых данных** (т.е. указать, какие параметры будут вычитываться, из какого контроллера или устройства, с каким интервалом вычитывать данные). Такая конфигурация задаётся программой "**Конфигуратор**" (Окно "Настройка архивов").

Запуск программы "[Менеджер хранилища](#)" производится автоматически при старте вычитывания, на компьютере, где находится хранилище.



В ПО "**SyTrack-WRT**" **ARC-ClientNet** предусмотрены следующие варианты лицензирования:

По количеству архивных дискретов/аналогов/счётчиков, которые необходимо собирать из контроллеров:

- до 100
- до 200
- до 300
- до 400
- от 400

По количеству сетевых серверов, с которых происходит сбор архивных данных:

- до 5
- от 5 до 10
- от 10

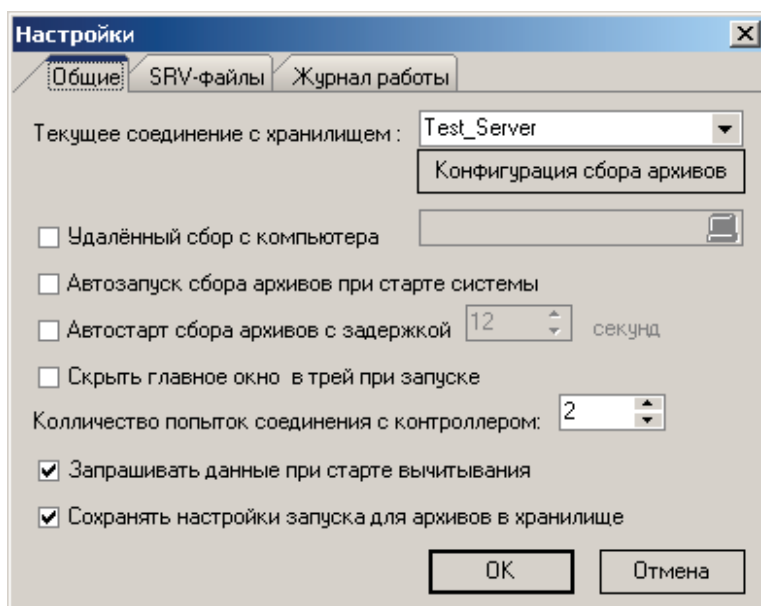
Также возможен следующий вариант лицензирования:

- сбор из хранилищ других производителей

6.4.1 Настройки программы

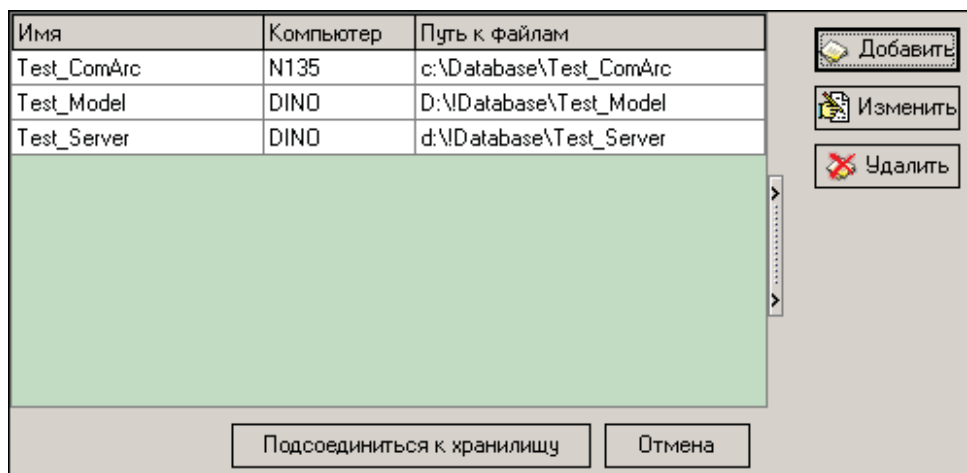
Перед запуском вычитывания архивов рекомендуется установить **настройки программы**.

Окно настроек программы можно вызвать, выбрав пункт меню "Параметры системы ->Конфигурация системы" или соответствующую кнопку на главном окне программы.



В данном окне можно установить:

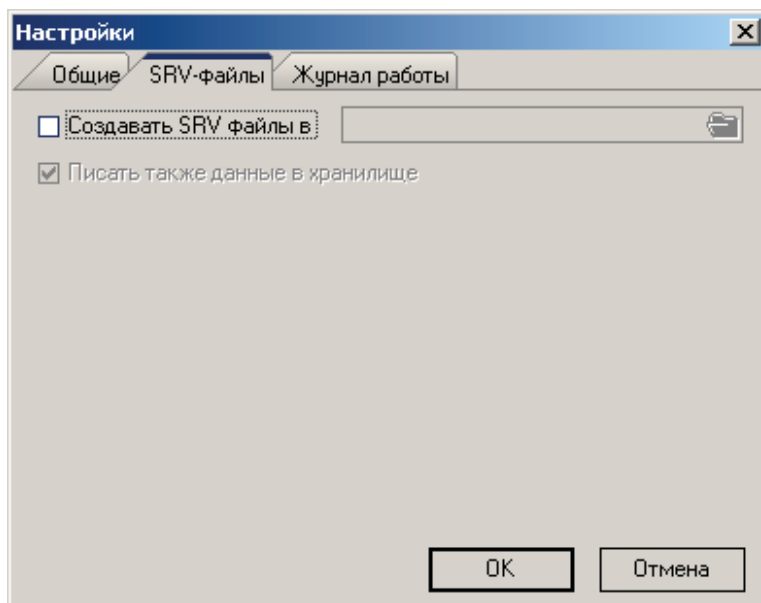
- **Текущее соединение с хранилищем**, в которое будет производиться вычитывание. При нажатии кнопки выпадающего списка появится стандартное окно выбора псевдонима хранилища. В данном окне можно как изменить выбранный псевдоним, так и добавить новый. Настройки псевдонима аналогичны настройкам при [создании нового хранилища](#).



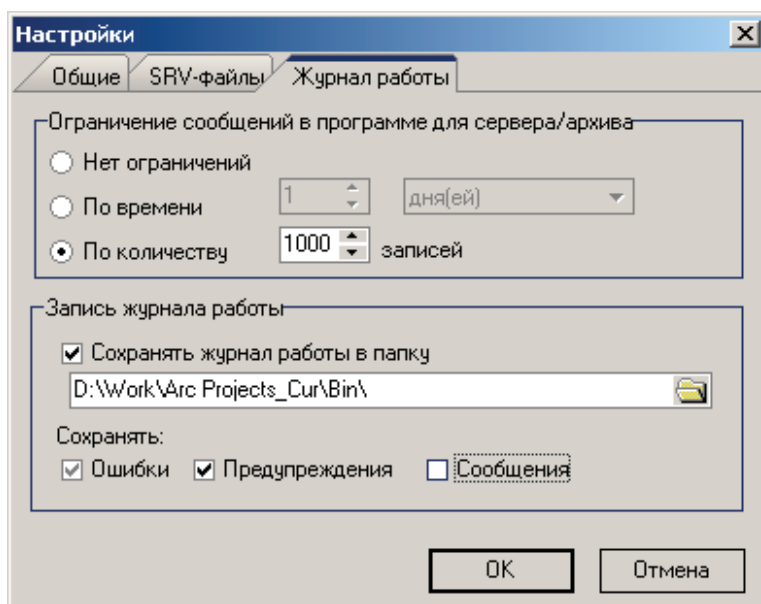
- **Удаленный сбор с компьютера**, с которого будет производиться сбор архивов (т.е. компьютер, на котором запущена программа WinDecont). Если название компьютера не указано, сбор будет производиться с локального компьютера.
- **Автозапуск сбора архивов при старте системы** позволяет запускать программу автоматически при старте операционной системы.
- **Автостарт сбора архивов с задержкой** (в секундах) позволяет установить интервал времени, по истечении которого после запуска программы будет автоматически произведён старт вычитывания архивов. Данная опция используется для синхронизации с другими программами.
- Опция **Скрыть главное окно в трей при запуске** - при запуске программы главное окно программы скрывается в трей.
- Опция **Количество попыток соединения с контроллером**, указывает сколько попыток соединения будет предпринято, прежде чем будет установлено ожидание соединения.
- Опция **Запрашивать данные при старте вычитывания** позволяет при запуске сбора данных не ожидая времени вычитывания данных установить соединение с сервером (имеется ввиду поставщик архивных данных) и запросить

данные. Данная опция не распространяется на архивы, у которых период вычитывания не привязан к астрономическому времени (например "пер_1мин"), т.е. для них при старте сбора всегда будет производиться запрос данных

- Опция **Сохранять настройки запуска для архивов в хранилище** позволяет сохранить в хранилище состояние запуска архива (запущено или приостановлено с него вычитывание).



- Опция **создавать SRV-файлы** предоставляет альтернативный формат для сохранения вычитанных данных (см. [Формат SRV-файлов](#)). При этом есть возможность также писать данные в хранилище (опция **Писать также данные в хранилище**). Если не указано создание SRV-файлов, то данные пишутся в хранилище.



- **Журнал работы.** Сообщения выдаваемые при сборе данных (см. [Текущее состояние сбора](#)) могут сохраняться на время работы программы в памяти, а также в файл. Данные сообщения позволяют более быстро выявить причину неправильной работы системы. Есть возможность ограничить количество сообщений, хранящихся в памяти для предотвращения ее утечки. Можно ограничить как по времени, так и по их количеству для каждого сервера и архива. Рекомендуется также сохранять в файл ошибки и предупреждения, а также, если требуется и сообщения. Файлы создаются в указанной папке на каждый день в текстовом формате, имеют названия в следующем формате DarxServer_DD-MM-YY.log.

6.4.2 Текущее состояние сбора

Для просмотра текущего состояния сбора в меню главного окна программы выберите <Параметры системы->Текущее состояние> или нажмите соответствующую кнопку на главном окне программы. Архивы, с которых производится сбор данных, сгруппированы по серверам(сервер определяется номером контролера, типом архива и номером экземпляра задачи в контролере).

Первый столбец с рисунком показывает старт/стоп сбора данных с сервера/архива:

Контроллер	Версия	Экземпляр	Состояние
2	<неизв.>	0	WinDecont запущен [20.12.05 13:57:25]
Архив №1	Аналоги с периодом <неизвестно>	Время данных: 20.12.05 13:56:50	<Неизвестно>
Архив №2	Аналоги с периодом <неизвестно>	Время данных: 20.12.05 13:56:55	<Неизвестно>
2	<неизв.>	0	WinDecont запущен [20.12.05 13:57:25]
Архив №1	Счётчики с периодом <неизвестно>	Время данных: 20.12.05 13:57:05	<Неизвестно>
2	<неизв.>	0	WinDecont запущен [20.12.05 13:57:25]
Архив №1	События	Время данных: 20.12.05 13:57:25	<Неизвестно>
Архив №2	События	Время данных: 20.12.05 13:57:26	<Неизвестно>
3	<неизв.>	0	WinDecont запущен [20.12.05 13:57:25]
Архив №1	Аналоги с периодом <неизвестно>	Время данных: 30.12.99 00:00:00	<Неизвестно>
3	<неизв.>	0	WinDecont запущен [20.12.05 13:57:25]
Архив №1	Счётчики с периодом <неизвестно>	Время данных: 30.12.99 00:00:00	<Неизвестно>

6.4.3 Вычитывание архивов

Контроллер	Версия	Экземпляр	Состояние
2	<неизв.>	0	WinDecont запущен [20.12.05 13:57:25]
Архив №1	Аналоги с периодом <неизвестно>	Время данных: 20.12.05 13:56:50	<Неизвестно>
Архив №2	Аналоги с периодом <неизвестно>	Время данных: 20.12.05 13:56:55	<Неизвестно>
2	<неизв.>	0	WinDecont запущен [20.12.05 13:57:25]
Архив №1	Счётчики с периодом <неизвестно>	Время данных: 20.12.05 13:57:05	<Неизвестно>
2	<неизв.>	0	WinDecont запущен [20.12.05 13:57:25]
Архив №1	События	Время данных: 20.12.05 13:57:25	<Неизвестно>
Архив №2	События	Время данных: 20.12.05 13:57:26	<Неизвестно>
3	<неизв.>	0	WinDecont запущен [20.12.05 13:57:25]
Архив №1	Аналоги с периодом <неизвестно>	Время данных: 30.12.99 00:00:00	<Неизвестно>
3	<неизв.>	0	WinDecont запущен [20.12.05 13:57:25]
Архив №1	Счётчики с периодом <неизвестно>	Время данных: 30.12.99 00:00:00	<Неизвестно>

Вычитывание архивов может производиться **постоянно** ([постоянное вычитывание](#)) - данные запрашиваются через временные интервалы согласно конфигурации вычитывания.

Конфигурацию вычитывания можно просмотреть, выбрав пункт меню "Параметры системы -> Настройки", в окне "Настройки" на закладке "Общие" нажать кнопку "Конфигурация сбора архивов".

При [выборочном вычитывании](#) производится вычитывание данных с определенного контроллера и за определенный интервал времени без ожидания.

6.4.3.1 Вычитывание архивов

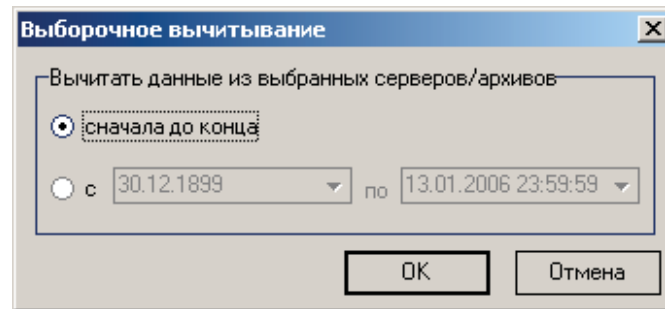
Управление вычитыванием архивов производится с помощью ряда кнопок главного окна приложения или выбором соответствующего пункта меню. В ходе вычитывания можно произвести рестарт вычитывания, если была изменена конфигурация вычитывания. Предусмотрена возможность в процессе вычитывания произвести вычитывание, не ожидая временного интервала, установленного в конфигурации вычитывания. Для этого выберите пункт меню "Чтение архивов -> Вычитать, не ожидая" или нажмите соответствующую кнопку на главном окне программы.

Контроллер	Версия	Экземпляр	Состояние
Архив №1	Аналоги с периодом 10 секунд(ы)	2	Соединение установлено [20.12.05 14:15:07]
Архив №2	Аналоги с периодом 5 секунд(ы)	0	Время данных: 20.12.05 14:15:20 Ожидание сетевого обмена до 20.12.05 14:16:01 [20.12.05 14:15:11]
Архив №1	Счётчики с периодом 5 секунд(ы)	2	Время данных: 20.12.05 14:15:35 Ожидание сетевого обмена до 20.12.05 14:16:01 [20.12.05 14:15:12]
Архив №1	События	2	Время данных: 20.12.05 13:57:25 <Неизвестно>
Архив №2	События	0	Время данных: 20.12.05 13:57:26 <Неизвестно>
Архив №1	Аналоги с периодом <неизвестно>	3	Ожидание соединения до 20.12.05 14:16:01 [20.12.05 14:15:07]
Архив №1	Счётчики с периодом <неизвестно>	3	Ожидание соединения до 20.12.05 14:16:01 [20.12.05 14:15:07]
Архив №1	Счётчики с периодом <неизвестно>	0	Время данных: 30.12.99 00:00:00 <Неизвестно>

Следить за ходом процесса вычитывания можно, выбрав в меню "Параметры системы -> Текущее состояние" или соответствующую кнопку на главном окне программы. В появившемся окне имеются пять закладок, в четырёх первых из которых отражается ход вычитывания архивов по их типу (события, аналогии, счётчики и внешние архивы). Закладки становятся видимыми, если в них есть какая-либо информация. В последней закладке выводятся ошибки, возникающие при вычитывании.

6.4.3.2 Выборочное вычитывание

В программе предусмотрена возможность **выборочного вычитывания архивов**. Архивы можно вычитать за определённый временной период и для определённых логических адресов контроллеров. Чтобы произвести выборочное вычитывание, выберите пункт меню "Чтение архивов -> Выборочное вычитывание". В появившемся окне укажите временной интервал, и из каких контроллеров производить вычитывание (возможно вычитывание из указанного одного или из всех контроллеров).



6.4.3.3 Вычитывание по сети SYSNET из другого хранилища

Сбор данных в другое хранилище возможен, если два компьютера объединены в сеть SYSNET, т.е. если с любого из компьютеров через программу WinDecont возможно установить связь с другим контроллером WinDecont на другом компьютере.

Компьютер, с которого собирают данные, назовём "сервером". Предполагается, что на нём происходит общий сбор данных с контроллеров или внешних устройств в "хранилище-сервер". Второй компьютер, который будет запрашивать данные по сети SYSNET, назовём "клиент", а хранилище, в которое он их будет записывать, назовём "хранилище-клиент".

На стороне клиента необходимо определить конфигурацию вычитывания архивов. Конфигурация определяется в окне "Справочники хранилища" (вызов из меню "Параметры системы ->Настройки", далее нажимаем кнопку "Конфигурация сбора архивов"), пункт "Архивы удаленного хранилища".

ID	Контроллер	Псевдоним удаленного хранилища	Префикс имени параметров																																																																											
1		2 Test_Arc_S	Contr2																																																																											
<table border="1"> <thead> <tr> <th>ID</th> <th>Тип архива</th> <th>Тип периода</th> <th>Период</th> <th>Период чтения</th> <th>Запускать</th> </tr> </thead> <tbody> <tr> <td>2</td> <td>Дискретны</td> <td>Не задается</td> <td></td> <td>0 час_5сек_1мин</td> <td><input checked="" type="checkbox"/></td> </tr> <tr> <td colspan="6"> <table border="1"> <thead> <tr> <th>ID</th> <th>Код параметра</th> <th>Код задачи</th> </tr> </thead> <tbody> <tr> <td>11</td> <td>2:D:1</td> <td>КФГ</td> </tr> <tr> <td>12</td> <td>2:D:2</td> <td>КФГ</td> </tr> <tr> <td>13</td> <td>2:D:3</td> <td>КФГ</td> </tr> </tbody> </table> </td> </tr> <tr> <td>1</td> <td>Аналоги</td> <td>Секунда</td> <td></td> <td>30 час_5сек_1мин</td> <td><input checked="" type="checkbox"/></td> </tr> <tr> <td colspan="6"> <table border="1"> <thead> <tr> <th>ID</th> <th>Код параметра</th> <th>Код задачи</th> </tr> </thead> <tbody> <tr><td>1</td><td>2:A:1</td><td>КФГ</td></tr> <tr><td>2</td><td>2:A:2</td><td>КФГ</td></tr> <tr><td>3</td><td>2:A:3</td><td>КФГ</td></tr> <tr><td>4</td><td>2:A:4</td><td>КФГ</td></tr> <tr><td>5</td><td>2:A:5</td><td>КФГ</td></tr> <tr><td>6</td><td>2:A:6</td><td>КФГ</td></tr> <tr><td>7</td><td>2:A:7</td><td>КФГ</td></tr> <tr><td>8</td><td>2:A:8</td><td>КФГ</td></tr> <tr><td>9</td><td>2:A:9</td><td>КФГ</td></tr> <tr><td>10</td><td>2:A:10</td><td>КФГ</td></tr> </tbody> </table> </td> </tr> </tbody> </table>				ID	Тип архива	Тип периода	Период	Период чтения	Запускать	2	Дискретны	Не задается		0 час_5сек_1мин	<input checked="" type="checkbox"/>	<table border="1"> <thead> <tr> <th>ID</th> <th>Код параметра</th> <th>Код задачи</th> </tr> </thead> <tbody> <tr> <td>11</td> <td>2:D:1</td> <td>КФГ</td> </tr> <tr> <td>12</td> <td>2:D:2</td> <td>КФГ</td> </tr> <tr> <td>13</td> <td>2:D:3</td> <td>КФГ</td> </tr> </tbody> </table>						ID	Код параметра	Код задачи	11	2:D:1	КФГ	12	2:D:2	КФГ	13	2:D:3	КФГ	1	Аналоги	Секунда		30 час_5сек_1мин	<input checked="" type="checkbox"/>	<table border="1"> <thead> <tr> <th>ID</th> <th>Код параметра</th> <th>Код задачи</th> </tr> </thead> <tbody> <tr><td>1</td><td>2:A:1</td><td>КФГ</td></tr> <tr><td>2</td><td>2:A:2</td><td>КФГ</td></tr> <tr><td>3</td><td>2:A:3</td><td>КФГ</td></tr> <tr><td>4</td><td>2:A:4</td><td>КФГ</td></tr> <tr><td>5</td><td>2:A:5</td><td>КФГ</td></tr> <tr><td>6</td><td>2:A:6</td><td>КФГ</td></tr> <tr><td>7</td><td>2:A:7</td><td>КФГ</td></tr> <tr><td>8</td><td>2:A:8</td><td>КФГ</td></tr> <tr><td>9</td><td>2:A:9</td><td>КФГ</td></tr> <tr><td>10</td><td>2:A:10</td><td>КФГ</td></tr> </tbody> </table>						ID	Код параметра	Код задачи	1	2:A:1	КФГ	2	2:A:2	КФГ	3	2:A:3	КФГ	4	2:A:4	КФГ	5	2:A:5	КФГ	6	2:A:6	КФГ	7	2:A:7	КФГ	8	2:A:8	КФГ	9	2:A:9	КФГ	10	2:A:10	КФГ
ID	Тип архива	Тип периода	Период	Период чтения	Запускать																																																																									
2	Дискретны	Не задается		0 час_5сек_1мин	<input checked="" type="checkbox"/>																																																																									
<table border="1"> <thead> <tr> <th>ID</th> <th>Код параметра</th> <th>Код задачи</th> </tr> </thead> <tbody> <tr> <td>11</td> <td>2:D:1</td> <td>КФГ</td> </tr> <tr> <td>12</td> <td>2:D:2</td> <td>КФГ</td> </tr> <tr> <td>13</td> <td>2:D:3</td> <td>КФГ</td> </tr> </tbody> </table>						ID	Код параметра	Код задачи	11	2:D:1	КФГ	12	2:D:2	КФГ	13	2:D:3	КФГ																																																													
ID	Код параметра	Код задачи																																																																												
11	2:D:1	КФГ																																																																												
12	2:D:2	КФГ																																																																												
13	2:D:3	КФГ																																																																												
1	Аналоги	Секунда		30 час_5сек_1мин	<input checked="" type="checkbox"/>																																																																									
<table border="1"> <thead> <tr> <th>ID</th> <th>Код параметра</th> <th>Код задачи</th> </tr> </thead> <tbody> <tr><td>1</td><td>2:A:1</td><td>КФГ</td></tr> <tr><td>2</td><td>2:A:2</td><td>КФГ</td></tr> <tr><td>3</td><td>2:A:3</td><td>КФГ</td></tr> <tr><td>4</td><td>2:A:4</td><td>КФГ</td></tr> <tr><td>5</td><td>2:A:5</td><td>КФГ</td></tr> <tr><td>6</td><td>2:A:6</td><td>КФГ</td></tr> <tr><td>7</td><td>2:A:7</td><td>КФГ</td></tr> <tr><td>8</td><td>2:A:8</td><td>КФГ</td></tr> <tr><td>9</td><td>2:A:9</td><td>КФГ</td></tr> <tr><td>10</td><td>2:A:10</td><td>КФГ</td></tr> </tbody> </table>						ID	Код параметра	Код задачи	1	2:A:1	КФГ	2	2:A:2	КФГ	3	2:A:3	КФГ	4	2:A:4	КФГ	5	2:A:5	КФГ	6	2:A:6	КФГ	7	2:A:7	КФГ	8	2:A:8	КФГ	9	2:A:9	КФГ	10	2:A:10	КФГ																																								
ID	Код параметра	Код задачи																																																																												
1	2:A:1	КФГ																																																																												
2	2:A:2	КФГ																																																																												
3	2:A:3	КФГ																																																																												
4	2:A:4	КФГ																																																																												
5	2:A:5	КФГ																																																																												
6	2:A:6	КФГ																																																																												
7	2:A:7	КФГ																																																																												
8	2:A:8	КФГ																																																																												
9	2:A:9	КФГ																																																																												
10	2:A:10	КФГ																																																																												

Рассмотрим более подробно таблицы конфигурации вычитывания.

В первом уровне таблицы определяются логические адреса контроллеров WinDecont, запущенных на стороне сервера, соответствующие псевдонимы хранилища, определённые также на стороне сервера и префикс к имени архивируемых параметров.

На втором уровне таблицы определяется список вычитываемых архивов с сервера. Для каждого архива необходимо определить тип архива (Аналоги, Счетчики, Дискретные, Аналогии по изменению или Счетчики по изменению), период архива (поля "Тип перехода" и "Период") и период чтения (запроса) данных.

На третьем уровне таблицы для каждого архива необходимо определить список вычитываемых параметров (коды параметров из справочника параметров).




Заданная конфигурация будет сохранена в хранилище на стороне клиента.

На стороне сервера требуется запустить программу "[Сервер хранилища](#)".

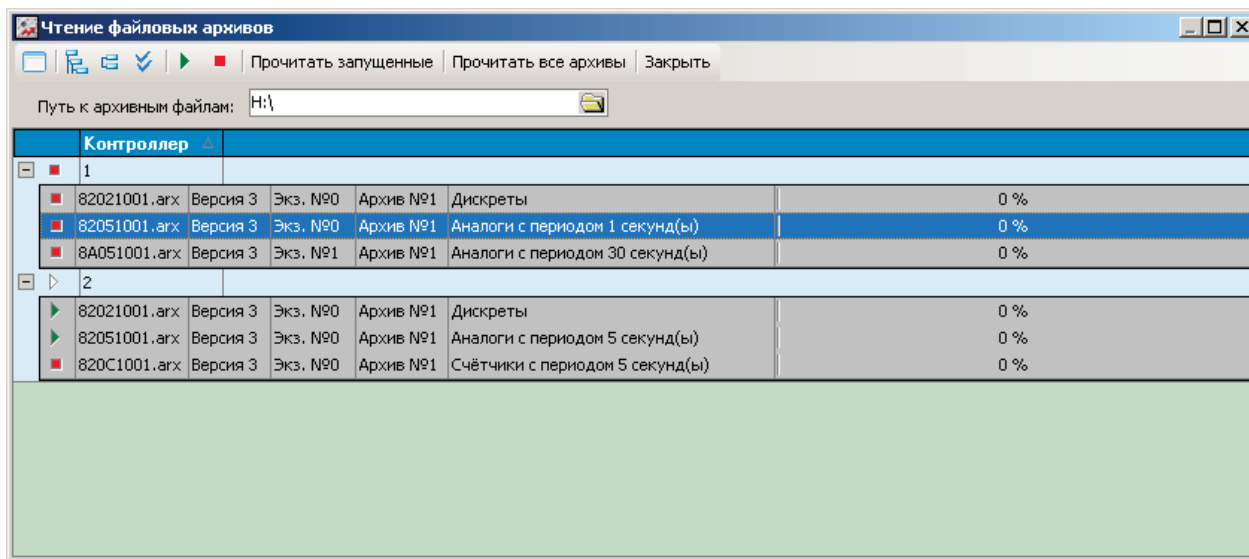
На стороне клиента запускается сбор архивов с описанной выше настройкой вычитывания архивов. В процессе получения данных будут созданы запрашиваемые параметры в [справочнике параметров](#) и архивы в соответствующих справочниках [периодических архивов](#) и [архивов по изменению](#). К именам параметров будет добавлен префикс, указанный в поле "Префикс имени параметра", а код параметра будет сформирован в следующем виде "S[Адрес контроллера]:[Код задачи]:[Код параметра]".

6.4.3.4 Чтение файловых архивов

Источником данных для данного способа вычитывания архивов являются файлы вида ???????.arx, формируемые контроллером Деконт А9 на внешний носитель при изменении системного параметра "Сохранить архивы на внешний носитель" (см. Справочник системных параметров). Возможно чтение с только стандартных архивов контроллеров Деконт А9 (компоненты "Архив аналогов", "Архив счётчиков" и "Архив событий").

Вызов окна чтения файловых архивов возможен через меню главного окна (Чтение архивов->Файловые архивы) или нажатием горячей клавиши F5. В данном окне требуется указать путь к архивным файлам, где находятся папки вида LA_?????. После выбора пути к файлам будет показана конфигурация найденных архивов для их последующего чтения и записи в хранилище. Если в хранилище есть конфигурация вычитывания стандартных архивов, то будет произведено сравнение с ней и вычитываемые архивы, которые в ней присутствуют будут помечены запущенными . Читать или нет архив можно установить кнопками  и  соответственно. Запустить чтение архивных файлов можно нажав кнопку "Прочитать запущенные", т.е. запустится чтение только для архивов помеченных к запуску, для запуска чтения всех архивов вне зависимости от пометки к запуску нужно нажать кнопку "Прочитать все архивы".

Чтение файловых архивов производится вне зависимости от общего запуска чтения архивов с контроллеров или устройств.



6.4.3.5 Формат SRV-файлов

В [настройках программы](#) указывается каталог создания SRV-файлов, в подкаталогах которого папки ArInt, ArEvent, ArCount - прочитанные архивные данные для аналогов, событий и счетчиков соответственно.

Для каждого периода усреднения (для аналогов) и астрономического интервала (для счетчиков) создается отдельный каталог с архивными данными. Название каталога совпадает с обозначением периода. Для архивов событий считается, что период равен 30-ти секундам. Приняты следующие обозначения периодов и интервалов архивов:

ЧЧ_ММ_СС часы, минуты, секунды (для интервалов меньших одного дня)
 Day период равен календарным суткам
 Month период равен календарному месяцу.

Например, расположение каталогов для архивов счетчиков:

<Рабочая директория>

\ArCount

\00_10_00 10 минут

\00_00_30 30 секунд

\01_30_00 один час тридцать минут

\Month календарный месяц

В каждом каталоге с архивными данными находятся следующие файлы:

- DataR.DAD файл с данными, готовыми к обработке внешним приложением. Сервер сбора не модифицирует и не использует данные из этого файла.
- ~DataSrv.DAD служебный файл сервера. Содержит текущие данные, читаемые по сети из контроллеров.

Поддерживается следующий алгоритм обмена:

- ✓ пока существует файл с готовыми данными, сервер новые данные записывает в служебный файл
- ✓ если файла с готовыми данными нет и служебный файл не пуст, тогда сервер формирует из служебного новый файл с готовыми данными.

.1 Формат файла с готовыми данными:

Служебная запись
Запись данных 1
Запись данных 2
....
Запись данных N

.2 Форматы записей

Служебная запись			
Название	Тип	Длина	Назначение
RSrvLen	word	2	Длина данной служебной записи
RType	word	2	Тип архивных данных
RVersion	word	2	Версия формата записи данных
RLen	word	2	Длина записи данных (в байтах)
RPeriod	DTime	4	Интервал архивирования данных
RPeriodType	word	2	Тип интервала архивирования (0)
резерв		18	
Итого		32	
Архив событий			
RType	69	E	
RVersion	1		
RLen	32		
Название	Тип		Назначение
LA	word	2	Логический адрес контроллера
ParmGNo	word	2	Номер параметра в базе контроллера
TimeEvent	DTime	4	Время события
TimeEventMs	word	2	Мсек события
State	word	2	Новое значение дискрета
InstNo	byte	1	Номер экземпляра компонента архива
TimeCreate	DTime	4	Время создания архива
RecordNo	dword	4	Номер записи в архиве
ArchiveNo	byte	1	Номер архива в компоненте
DTNow	TDateTme	8	Время формирования данной записи в файле
	резерв	2	резерв не используется
Длина записи		32	
Архив счетчиков			
RType	67	C	
RVersion	1		
RLen	32		
Название	Тип		Назначение
LA	word	2	Логический адрес контроллера
ParmGNo	word	2	Номер параметра в базе контроллера
TimeEvent	DTime	4	Время счетчика

Count	dword	4	Значение счетчика
InstNo	byte	1	Номер экземпляра компонента архива
TimeCreate	DTime	4	Время создания архива
RecordNo	dword	4	Номер записи в архиве
DTNow	TDateTime	8	Время формирования данной записи в файле
	резерв	3	резерв не используется
Длина записи		32	
Архив аналогов			
RType	73	I	
RVersion	1		
RLen	32		
Название	Тип		Назначение
LA	word	2	Логический адрес контроллера
ParmGNo	word	2	Номер параметра в базе контроллера
TimeEvent	DTime	4	Время измерения
Value	float	4	Сумма измерений
Count	word	2	Количество измерений
InstNo	byte	1	Номер экземпляра компонента архива
TimeCreate	DTime	4	Время создания архива
RecordNo	dword	4	Номер записи в архиве
DTNow	TDateTime	8	Время формирования данной записи в файле
	резерв	1	резерв не используется
Длина записи		32	

.3 Форматы данных

Название	Тип	Байт		
DTime	dword	4		количество секунд с 1980 года
Время в сек				
State	word	2		Формат в битах
Состояние параметра				U D R x xxxx xxxx xxxx
				U достоверность
				1-значение параметра не определено
				D бит динамики
				R резервный бит
				x xxxx xxxx xxxx
				код ошибки при U=1
				код значения дискрета при U=0
DTNow	TDateTime	8		Object Pascal TDateTime

Замечания.

Поля TimeCreate и RecordNo предназначены для отслеживания целостности архивных данных. Их анализ позволяет фиксировать переформатирование архива в контроллере.

Поле TimeEvent содержит кодированное значение времени, к которому относится данная запись. Для архива событий это количество 5 секундных интервалов, прошедших с начала суток. Значение данного поля следует рассматривать как без знаковое (всегда положительное) целое число.

Поле TimeEventMs конкретизирует, указанное в TimeEvent время. Это количество миллисекунд, которые необходимо добавить к TimeEvent. Значение данного поля может изменяться от 0 до 4999 (4 сек 999 мсек).

Поле ParmNo следует рассматривать как беззнаковое целое число. Если его значение меньше чем 32768, то это номер дискрета в базе текущих значений дискретов в контроллере. Иначе, это запись о системном событии:

32769 – старт контроллера

32770 – останов контроллера

32771 – перевод астрономического времени в контроллере (FState = ON –начало перевода времени, OFF – конец перевода времени).

6.5 Просмотр архивов

ПО "SyTrack-WRT" ARC-View (программа "**Просмотр архивных данных**") предназначено для обработки, отображения, печати и экспорта в текстовом и графическом виде архивных данных, расположенных в хранилище.

Программа поддерживает многооконный интерфейс, автоматическое дочитывание данных обработки, отображение, печать и экспорт в текстовом и графическом виде архивных данных, расположенных в хранилище.

Основные функции программы "Просмотр архивных данных":

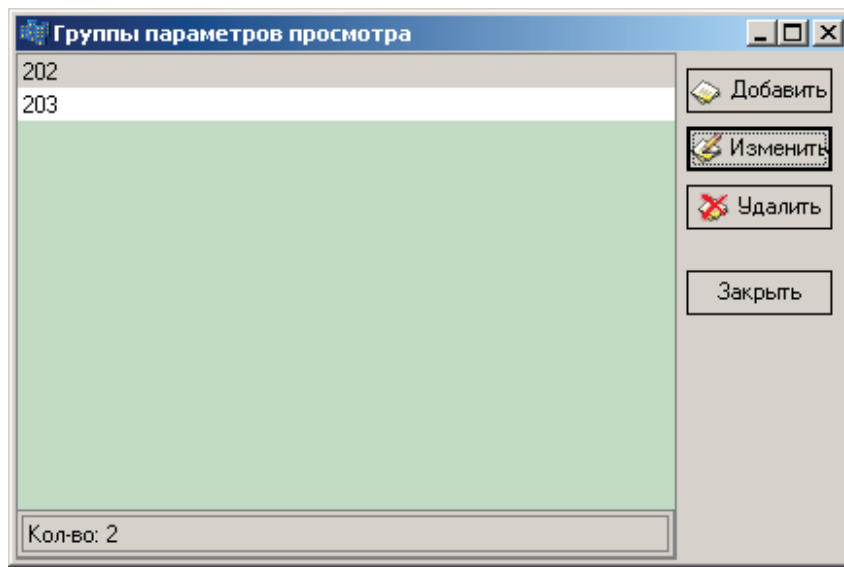
- представление архивных данных в табличном и графическом виде (аналоги, счетчики, дискреты, аналоги по изменению, счетчики по изменению, периодичные архивы, архивы по изменению, оперативный журнал, графики);
- обработка полученных архивных данных;
- печать архивных данных в табличном и графическом виде;
- экспорт данных (формат: xls, html, xml и txt.)

6.5.1 Соединение с хранилищем

При запуске программы производится автоматическое подсоединение к хранилищу, с которым в последний раз удачно было установлено соединение. Для установления соединения с другим хранилищем выберите пункт меню "Настройка -> Общие настройки" или нажмите соответствующую кнопку на главной форме. В окне настроек перейдите на закладку "Общие" и нажмите кнопку выбора псевдонима хранилища. Появится окно аналогичное подсоединению к хранилищу через "Менеджер хранилища" (см. [Соединение с хранилищем и его обслуживание](#)).

6.5.2 Группы параметров просмотра

Для классификации параметров и удобства их просмотра в программе реализованы "**группы параметров просмотра**". Для их добавления, изменения и удаления выберите пункт меню "Настройка -> Группы параметров" или нажмите соответствующую кнопку на главной форме. В появившемся окне будет выведен список групп (если они созданы) и ряд кнопок для их редактирования. Имеется возможность работать с группами в виде иерархического дерева. Для этого нажмите кнопку в левом верхнем углу или <Ctrl> + <M>.



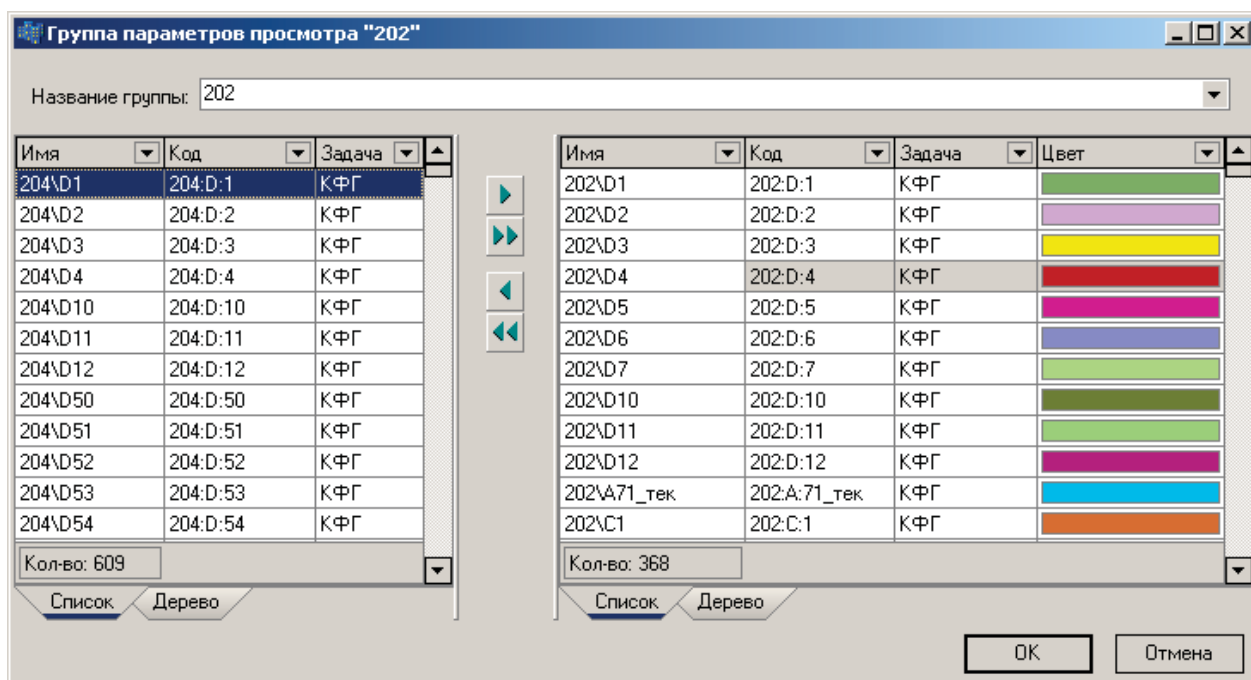
При добавлении или изменении группы в появившемся окне нужно указать имя группы и набор параметров, входящих в эту группу. Имя группы должно быть уникальным в хранилище и содержать не более 127 символов. Список всех параметров можно просматривать также в виде дерева. Для этого в верхнем левом углу нажмите соответствующую кнопку или <Ctrl> + <M>.

Для добавления и исключения параметров из группы имеется ряд кнопок, расположенных между списками. Кнопка ">" служит для включения в группу выделенных параметров из списка всех параметров. Кнопка "<" предназначена для исключения параметров из группы. Для выделения ряда идущих друг за другом параметров используйте клавишу <Shift>, для выделения отдельных параметров – клавишу <Ctrl>.

Кнопки ">>" и "<<" предназначены соответственно для включения всех параметров в группу и исключения всех параметров из группы. При этом выделение параметров в обоих списках значения не имеет. В группе обязательно должен присутствовать хотя бы один параметр.

При просмотре в табличном виде интегральных архивов и легенды графиков (см. [Просмотр интегральных архивов](#)) имеет значение порядок параметров в группе. Для упорядочивания параметров имеются две кнопки "стрелка вверх" и "стрелка вниз", которые соответственно перемещают выбранный параметр вверх или вниз в списке.

При просмотре графиков, чтобы отличить цвет одного графика параметра от другого, для каждого параметра в группе можно определить цвет (щелчок левой кнопки мыши на строке выбранного параметра). Цвет одного и того же параметра в разных группах может быть различным.



6.5.3 Просмотр данных

Просмотр архивных данных может осуществляться из определённого типа архива, по выбранным параметрам и за указанный интервал времени (см. [Параметры просмотра данных](#)).

Просмотр данных осуществляется в табличном (текстовом) и в графическом виде.

Возможен просмотр:

- [периодических архивов](#) (архив периодических аналогов и периодических счетчиков);
- [архивов по изменению](#) (архив событий, архив аналогов по изменению и архив счетчиков по изменению);
- [оперативного журнала](#);
- [графиков](#)

6.5.3.1 Параметры просмотра данных

Для просмотра архивных данных нужно выбрать пункт меню "Просмотр данных -> Новое окно просмотра данных" или соответствующую кнопку на главной форме. В появившемся окне "Параметры просмотра данных" нужно указать:

- **временной интервал просмотра;**
- **группу просмотра;**
- **период данных;**
- **в каком виде и какие данные показывать.**

Параметры просмотра данных

Временной интервал просмотра

За период с 01.12.2005 по 19.12.2005 23:59:59

За последний(ие) 1

За указанный месяц : 2004 года

Группа и период данных

Группа параметров : Все параметры

Период данных : 30 минута(ы)

Фильтр просмотра

Аналогии Аналогии по изменению Графики

Счётчики Счётчики по изменению

Дискретны Оперативный журнал

OK Отмена

Временной интервал просмотра можно указать тремя способами:

- за указанный период;
- за последнее определенное количество секунд, минут, часов, дней, месяцев, кварталов и годов;
- за выбранный месяц определенного года;

При этом если выбрано время начала и конца просмотра, то установленное время начала просмотра относится к первому дню просмотра, а установленное время конца просмотра - к последнему дню. Если время начала и конца просмотра не выбрано, то время начала, относящееся к первому дню - "00:00:00", к последнему - "23:59:59".

Группу параметров можно выбрать в выпадающем списке созданных групп (см. [Группы параметров просмотра](#)). Помимо всех пользовательских групп, в списке есть группа "Все параметры", при выборе которой производится вычитывание всех значений параметров, имеющихся в хранилище. Добавить новую группу или изменить параметры в группе можно здесь же, нажав на кнопку справа от выпадающего списка групп (см. [Группы параметров просмотра](#)).

Период данных также выбирается в выпадающем списке, который формируется из периодов архивов, указанных в справочнике архивов (см. [Справочник архивов](#)). Для вычитывания только дискретов из выбранной группы нужно выбрать период данных "<неизвестно>".

Фильтр просмотра позволяет также сузить область вычитываемых и отображаемых данных и уменьшить время их вычитывания.

После определения параметров просмотра архивных данных появляется окно непосредственного их просмотра с возможными закладками:

- Аналоги;
- Счетчики;
- События;
- Аналоги по изменению;
- Счетчики по изменению;
- Оперативный журнал;
- Графики

6.5.3.2 Просмотр периодических архивов

Для просмотра интегральных архивов выберите закладку "Интегральные архивы". На данной закладке имеется ряд кнопок и таблица с полученными архивными данными за выбранный временной интервал с указанной группой и периодом данных. Данные выводятся в таблицу по мере их вычитывания из хранилища. При построении таблицы для каждого параметра выделяется свой собственный столбец. После вывода всех данных столбцы устанавливаются в том порядке, в каком описаны параметры в группе ([Группы параметров просмотра](#)).

Просмотр архивов - "Сагов" - [Группа "Все параметры" с периодом 30 минута(ы) с 01.12.05 00:00:00 по 19...]

Просмотр данных Настройка Экспорт Окна Справка

Аналоги Графики

Дата и время	202\У2\Мощность	202\У2\Мощность	202\У2\Мощность	202\У2\Мощность
01.12.05 00:00	232.000	0.000	0.000	1.000
01.12.05 00:30	229.600	0.000	0.200	0.800
01.12.05 01:00	228.800	0.000	0.000	1.000
01.12.05 01:30	229.400	0.000	0.000	1.000
01.12.05 02:00	229.800	0.000	0.000	0.800
01.12.05 02:30	230.400	0.000	0.000	1.000
01.12.05 03:00	230.800	0.000	0.000	1.000
01.12.05 03:30	230.600	0.000	0.200	1.000
01.12.05 04:00	230.400	0.000	0.000	0.800
01.12.05 04:30	230.200	0.000	0.000	1.000
01.12.05 05:00	229.400	0.000	0.000	1.000
01.12.05 05:30	228.800	0.000	0.000	0.800
01.12.05 06:00	226.400	0.000	0.200	1.000
01.12.05 06:30	226.000	0.000	0.000	1.000
01.12.05 07:00	222.400	0.000	0.000	0.800
01.12.05 07:30	219.800	0.000	0.000	1.000
01.12.05 08:00	219.200	0.000	0.000	0.800
01.12.05 08:30	215.600	0.000	0.000	0.800
01.12.05 09:00	212.200	0.000	0.200	0.800
01.12.05 09:30	213.400	0.000	0.000	1.000
01.12.05 10:00	218.200	0.000	0.000	0.800
01.12.05 10:30	217.000	0.000	0.000	0.800
01.12.05 11:00	217.800	0.000	0.000	1.000
01.12.05 11:30	219.800	0.000	0.000	0.800
01.12.05 12:00	221.200	0.000	0.000	1.000
01.12.05 12:30	220.400	0.000	0.200	0.800
Кол-во: 891				

Назначение кнопок (слева направо) на верхней панели окна:

- "Обновить данные" - повторное вычитывание с теми же параметрами просмотра;
- "Изменить параметры просмотра" - появляется окно, где можно изменить параметры просмотра и потом снова вычитать данные в это же окно;

Для каждого столбца в таблице с данными возможна их сортировка по возрастанию и убыванию их значений. Для этого нужно нажать на соответствующий столбец таблицы, по которому данные должны быть отсортированы. Изображаемая стрелка при этом показывает направление возрастания значений.

Внизу окна выводится строка с общим количеством значений и строк в таблице.

6.5.3.3 Просмотр архивов по изменению

Для просмотра архива дискретов выберите закладку "Архив дискретов". На данной закладке имеется ряд кнопок и таблица с полученными архивными данными за выбранный временной интервал, с указанной группой и периодом данных. Данные выводятся в таблицу по мере их вычитывания из хранилища.

Дата и время	Название параметра	Значение	Д	Расшифровка состояния	Информация об ошибке
16.11.05 18:21:14.842	302\D2	Не опред.			Канальное рассоединение [Код 771]
16.11.05 18:23:18.520	302\D2	1		Включено	
16.11.05 18:31:58.082	303\D2	Не опред.			Нет сессии [Код 770]
16.11.05 18:32:00.851	303\D2	Не опред.			В узле назначения нет такого сервера [Код 771]
16.11.05 18:33:58.871	303\D2	Не опред.			Узел на пути не отвечает [Код 794]
16.11.05 18:36:00.039	303\D2	Не опред.			В узле назначения нет такого сервера [Код 771]
16.11.05 18:36:06.695	203\Останов контроллера	16		Рестарт по запросу верхнего у	
16.11.05 18:36:24.630	203\Запуск контроллера	1		Запуск контроллера	
16.11.05 18:36:24.768	203\D10	Не опред.			Тайм-аут [Код 6]
16.11.05 18:36:27.207	203\Останов контроллера	16		Рестарт по запросу верхнего у	
16.11.05 18:38:00.165	303\D2	1		Включено	
16.11.05 18:39:01.544	203\Запуск контроллера	1		Запуск контроллера	
16.11.05 18:39:01.677	203\D10	Не опред.			Тайм-аут [Код 6]
16.11.05 18:39:04.117	203\Останов контроллера	16		Рестарт по запросу верхнего у	
16.11.05 18:39:50.536	203\Запуск контроллера	1		Запуск контроллера	
16.11.05 18:39:50.668	203\D10	Не опред.			Тайм-аут [Код 6]
16.11.05 18:39:53.116	203\Останов контроллера	16		Рестарт по запросу верхнего у	
16.11.05 18:42:42.874	203\Запуск контроллера	1		Запуск контроллера	
16.11.05 18:42:43.007	203\D10	Не опред.			Тайм-аут [Код 6]
16.11.05 18:42:45.859	203\D2	Не опред.			Тайм-аут [Код 6]
16.11.05 18:42:48.929	203\D3	Не опред.			Тайм-аут [Код 6]
16.11.05 18:42:52.034	203\D4	Не опред.			Тайм-аут [Код 6]
16.11.05 18:42:56.193	203\D1	1	*	Включено	
17.11.05 14:46:59.000	202\У2\Включение/Выключение электр	0		Отключено	
17.11.05 14:47:14.260	202\D2	Не опред.			Тайм-аут [Код 6]
17.11.05 14:49:01.463	203\D1	Не опред.			Тайм-аут [Код 6]
17.11.05 15:04:42.000	202\У2\Включение/Выключение электр	1		Включено	
17.11.05 15:18:45.000	202\У2\Включение/Выключение электр	0		Отключено	
17.11.05 15:20:26.000	202\У2\Включение/Выключение электр	1		Включено	

Кол-во: 779

В отличие от таблицы интегральных архивов значения дискретов выводятся не по столбцам, а по строкам.

В столбце "Дата и время" к времени через двоеточие приписываются миллисекунды вычитывания дискрета.

В столбце "Расшифровка состояния" поясняется значение состояния дискрета, если для параметра определена таблица расшифровки дискрета (см. [Справочник параметров](#)). Если состояние дискрета "Не опред." (Не определено), то выдаётся в столбце "Информация об ошибке" расшифровка ошибки и её код. Расшифровку ошибок можно экспортировать из файла *.db (файлы таблиц PARADOX). Чтобы это сделать, выберите пункт меню "Настройка -> Общие настройки" и нажмите на закладку "Архив дискретов". Нажав на кнопку <Обновить кодировку ошибок>, выберите соответствующий db-файл. На этой же закладке также можно указать, в каком виде выводить код ошибки: в десятичном или шестнадцатеричном виде.

Назначение кнопок на верхней панели окна такое же, как и для закладки "Интегральные архивы", за исключением двух

добавленных кнопок:

- "Установить/снять фильтр по именам параметров" - в появившемся окне можно снять выбор с дискретов или, наоборот, отметить только дискреты, которые должны присутствовать в данной выборке;
- "Установить/снять фильтр неопределённости" - фильтр скрывает все неопределённые состояния дискретов, а также убирает повторяющиеся состояния дискрета;
- Установить/снять фильтр тревоги - фильтр скрывает состояния дискретов, в которых устанавливается бит тревоги (динамики - столбец - "Д").

Так же, как и для интегральных архивов, возможна сортировка по каждому столбцу в таблице.

Общее количество записей в таблице выводится внизу окна.

6.5.3.4 Просмотр оперативного журнала

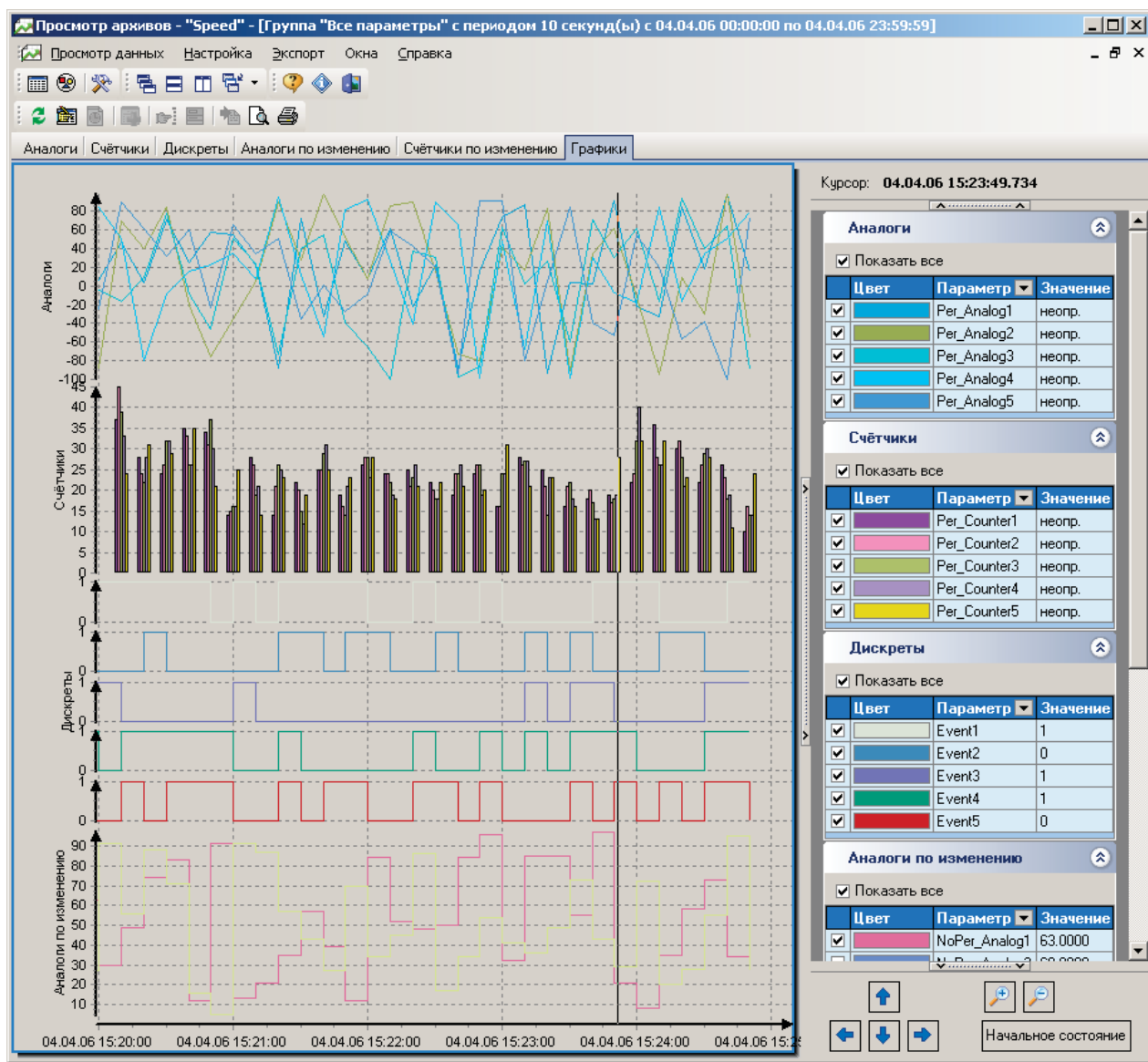
Для просмотра оперативного журнала при установке [параметров просмотра](#) данных в фильтре просмотра должен быть выбран оперативный журнал. Если задано автодочитывание данных, то данные выводятся в таблицу по мере их вычитывания из хранилища.

Дата и время	Действие	Объе	Диспетче	Комм	Квити	Время	Дисп	К:	Имя к
12.01.06 18:43:08.000	[22]:К.З цепи правого предела шибера	13-3(10)		АСУК ДЭГ	<input type="checkbox"/>			51	<Имя не э
12.01.06 18:43:08.000	[22]:К.З цепи правого предела шибера	13-2(9)		АСУК ДЭГ	<input type="checkbox"/>			51	<Имя не э
12.01.06 18:43:08.000	[22]:К.З цепи правого предела шибера	13-1(8)		АСУК ДЭГ	<input type="checkbox"/>			51	<Имя не э
12.01.06 18:43:08.000	[22]:К.З цепи правого предела шибера	Ю.3(7)		АСУК ДЭГ	<input type="checkbox"/>			51	<Имя не э
12.01.06 18:43:08.000	[22]:К.З цепи правого предела шибера	Ю.3-2(6)		АСУК ДЭГ	<input type="checkbox"/>			51	<Имя не э
12.01.06 18:43:08.000	[22]:К.З цепи правого предела шибера	1(5)		АСУК ДЭГ	<input type="checkbox"/>			51	<Имя не э
12.01.06 18:43:08.000	[22]:К.З цепи правого предела шибера	Ш.3(4)		АСУК ДЭГ	<input checked="" type="checkbox"/>			51	<Имя не э
12.01.06 18:43:08.000	[22]:К.З цепи правого предела шибера	2(3)		АСУК ДЭГ	<input type="checkbox"/>			51	<Имя не э
12.01.06 18:43:08.000	[22]:К.З цепи правого предела шибера	Ш1 бис(2)		АСУК ДЭГ	<input type="checkbox"/>			51	<Имя не э
12.01.06 18:43:08.000	[22]:К.З цепи правого предела шибера	Ш1(1)		АСУК ДЭГ	<input type="checkbox"/>			51	<Имя не э
12.01.06 18:43:09.000	[23]:К.З цепи сбрасывающего щитка	10-1(20)		АСУК ДЭГ	<input type="checkbox"/>			51	<Имя не э
12.01.06 18:43:09.000	[23]:К.З цепи сбрасывающего щитка	ЮВ-4(19)		АСУК ДЭГ	<input type="checkbox"/>			51	<Имя не э
12.01.06 18:43:09.000	[23]:К.З цепи сбрасывающего щитка	ЮВ-3(18)		АСУК ДЭГ	<input type="checkbox"/>			51	<Имя не э
12.01.06 18:43:09.000	[23]:К.З цепи сбрасывающего щитка	ЮВ-2(17)		АСУК ДЭГ	<input type="checkbox"/>			51	<Имя не э
12.01.06 18:43:09.000	[23]:К.З цепи сбрасывающего щитка	ЮВ-1(16)		АСУК ДЭГ	<input type="checkbox"/>			51	<Имя не э
12.01.06 18:43:09.000	[23]:К.З цепи сбрасывающего щитка	13-7(14)		АСУК ДЭГ	<input type="checkbox"/>			51	<Имя не э
12.01.06 18:43:09.000	[23]:К.З цепи сбрасывающего щитка	13-6(13)		АСУК ДЭГ	<input type="checkbox"/>			51	<Имя не э
12.01.06 18:43:09.000	[23]:К.З цепи сбрасывающего щитка	13-5(12)		АСУК ДЭГ	<input type="checkbox"/>			51	<Имя не э
12.01.06 18:43:09.000	[23]:К.З цепи сбрасывающего щитка	13-4(11)		АСУК ДЭГ	<input type="checkbox"/>			51	<Имя не э
12.01.06 18:43:10.000	[24]:К.З манометра управления минимально	9с-6(30)		АСУК ДЭГ	<input type="checkbox"/>			51	<Имя не э
12.01.06 18:43:10.000	[24]:К.З манометра управления минимально	9с-5(29)		АСУК ДЭГ	<input type="checkbox"/>			51	<Имя не э
12.01.06 18:43:10.000	[24]:К.З манометра управления минимально	9с-4(28)		АСУК ДЭГ	<input type="checkbox"/>			51	<Имя не э
12.01.06 18:43:10.000	[24]:К.З манометра управления минимально	9с-3(27)		АСУК ДЭГ	<input type="checkbox"/>			51	<Имя не э
12.01.06 18:43:10.000	[24]:К.З манометра управления минимально	9с-2(26)		АСУК ДЭГ	<input type="checkbox"/>			51	<Имя не э
12.01.06 18:43:10.000	[24]:К.З манометра управления минимально	9с-1(25)		АСУК ДЭГ	<input type="checkbox"/>			51	<Имя не э
12.01.06 18:43:10.000	[24]:К.З манометра управления минимально	10-3(23)		АСУК ДЭГ	<input type="checkbox"/>			51	<Имя не э
12.01.06 18:43:10.000	[24]:К.З манометра управления минимально	10-2(21)		АСУК ДЭГ	<input type="checkbox"/>			51	<Имя не э
Кол-во: 166									

6.5.3.5 Просмотр графиков

Для просмотра графиков выберите закладку "Графики". На данной закладке изображены графики всех вычитываемых параметров, а также их текущие значения (легенда параметров) в соответствии с курсором мыши на графике по горизонтальному срезу.

Часть графика не отображается, если время между поступающими данными больше их периода (для интегральных архивов), или состояние в данное время не определено (для архива дискретов).



Для увеличения масштаба нужно выбрать масштабируемую область графиков путём её выделения левой кнопкой мыши от левого верхнего (или нижнего) угла к правому нижнему (или верхнему).

Для движения графиков нужно, удерживая правую кнопку мыши, переместить курсор мыши на сдвигаемое расстояние в

нужном направлении.

Для восстановления графиков в начальное состояние выделите любую область, также как и при увеличении, но уже от правого верхнего (или нижнего) угла к левому верхнему (или нижнему).

Для скрытия и отображения нужного графика вновь имеется соответствующее поле в легенде параметров рядом с его цветовым отображением. При состоянии "Отмечено" график виден, иначе – скрыт. Также аналогично можно скрыть/показать группу графиков по типу архива.

Изменение цвета и толщины линий сетки графиков, а также цвета фона графиков и их толщины линий возможно на закладке "Графики" окна "Общие настройки" при выборе пункта меню "Настройка -> Общие настройки".

6.5.3.6 Автодочитывание данных

В программе реализована возможность **автоматического дочитывания данных** через определённый интервал времени.

Для установки интервала дочитывания выберите пункт меню "Настройка -> Общие настройки", закладку "Хранилище".

Автодочитывание данных происходит по всем открытым окнам с данными. Добавляются только новые данные, метка времени которых больше самой последней метки до автодочитывания.

6.5.4 Печать данных

Программа позволяет выводить на печать таблицы с данными, а также графики.

6.5.4.1 Печать таблиц и графиков

Перед непосредственной печатью рекомендуется вызвать предварительный просмотр распечатываемых данных. Для этого выберете пункт меню "Экспорт данных -> Предварительный просмотр". Печать можно произвести непосредственно из предварительного просмотра или выбрав пункт меню "Экспорт данных -> Печать...".

Print Preview

File View Format Go

Margins Left: 12,7 mm Top: 12,7 mm Right: 12,7 mm Bottom: 12,7 mm Header: 6,4 mm Footer: 6,4 mm

78%

Хранилище: "Speed"
Архив: "Счётчики"

Группа "Все параметры" с периодом 10 секунд(ы) с 19.02.06 00:00:00 по 19.02.06 23:59:59

Дата захвата	Per Counter1	Per Counter2	Per Counter3	Per Counter4	Per Counter5
19.02.06 14:20:00	7	9	19	29	38
19.02.06 14:20:10	44	54	58	42	42
19.02.06 14:20:20	72	78	80	90	93
19.02.06 14:20:30	94	104	112	122	122
19.02.06 14:20:40	131	137	138	148	157
19.02.06 14:20:50	145	148	175	178	178
19.02.06 14:21:00	179	183	191	194	203
19.02.06 14:21:10	207	209	210	215	217
19.02.06 14:21:20	221	230	234	240	240
19.02.06 14:21:30	243	250	251	252	259
19.02.06 14:21:40	248	275	280	283	284
19.02.06 14:21:50	287	291	294	304	307
19.02.06 14:22:00	313	319	322	327	335
19.02.06 14:22:10	337	343	344	344	353
19.02.06 14:22:20	342	344	347	372	374
19.02.06 14:22:30	384	384	385	390	394
19.02.06 14:22:40	403	408	409	414	417
19.02.06 14:22:50	427	434	435	435	437
19.02.06 14:23:00	443	450	459	459	448
19.02.06 14:23:10	471	474	484	484	488
19.02.06 14:23:20	494	497	500	509	512
19.02.06 14:23:30	512	518	522	527	528
19.02.06 14:23:40	530	538	539	540	541
19.02.06 14:23:50	549	555	557	559	549
19.02.06 14:24:00	571	579	588	599	601
19.02.06 14:24:10	607	607	615	625	633
19.02.06 14:24:20	637	639	643	644	654
19.02.06 14:24:30	659	645	672	674	684
19.02.06 14:24:40	685	688	690	695	695
19.02.06 14:24:50	695	704	704	709	719
19.02.06 14:25:00					

Page: 1 of 1 Pages Paper Size: 215,9 mm x 279,4 mm Status: Ready

Печать таблиц происходит с полным сохранением порядка следования данных, т.е. учитывается сортировка данных в столбце, а также его положение в группе.

При печати графиков также полностью соблюден принцип "как просматриваем, так и печатаем" (см. [Просмотр графиков](#)).

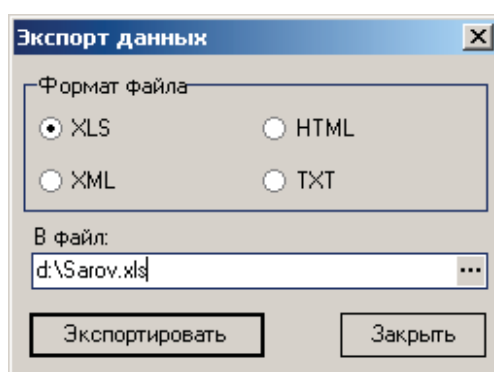
6.5.5 Экспорт данных

В программе возможен экспорт архивных данных в четыре формата: xls, html, xml и txt. Для экспорта не требуется установки MS Office или какого-либо другого программного обеспечения. Экспорт производится выбором пункта меню "Экспорт данных - > Экспорт данных" или нажатием соответствующей кнопки на главной форме.

В появившемся окне нужно указать:

- Формат файла XLS, HTML, XML или TXT.
- Имя и расположение файла, куда будут экспортированы данные (расширение файла можно не указывать).

Внимание! При экспорте данные в уже созданных файлах перетираются.



6.5.6 Запуск из внешней программы

Для запуска программы "Просмотр архивов" из внешней программы рекомендуется использовать функцию:

```
HINSTANCE ShellExecute(  
    HWND hwnd,  
    LPCTSTR lpOperation,  
    LPCTSTR lpFile,  
    LPCTSTR lpParameters,  
    LPCTSTR lpDirectory,  
    INT nShowCmd  
);
```

Эта функция возвращает значение >32 в случае успешного выполнения и значение ≤ 32 в случае ошибки.

Параметры:

hwnd дескриптор родительского окна

lpOperation строка, описывающая операцию: "open" - открытие (запуск), "print" - печать, "explore" - открыть окно с заданной папкой

lpFile имя файла

lpParameters параметры командной строки

lpDirectory текущая директория для запускаемой программы

nShowCmd параметр, указывающий, как будет показано приложение при открытии. Имеет тот же смысл, что и nCmdShow в WinMain

Пример запуска "Просмотра архивов":

```
if((UINT)ShellExecute(  
    NULL,  
    "open",  
    "DarxView.exe",  
    "-show",  
    NULL,  
    SW_SHOWNORMAL) <= 32) {  
    // обработка ошибок  
}
```

В примере в качестве командной строки указано "-show", что позволяет запускать одну копию приложения и делать активным окно уже запущенной копии.

6.6 Сервер хранилища

Программа **"Сервер хранилища"** предоставляет данные из хранилища по их запросу "Сбором архивов" через сеть SYSNET, т.е. если с любого из компьютеров через программу WinDecont возможно установить связь с другим контроллером WinDecont на другом компьютере.

"Сервер хранилища" должен быть запущен на компьютере вместе с программой "WinDecont", через который возможно установить соединение с удаленной программой "WinDecont", которая в свою очередь устанавливает связь с программой "Сбор архивов". Программа не требует каких-либо настроек, показывает текущее количество клиентов и их состояние.

Сервер хранилища		
Текущее состояние		
ID клиента	Компьютер	Псевдоним
35178872	KL-ALEX	Test_Arc_S
Отправлено 10 записей		[30.03.06 16:28:06]
Отправлено 3 записей		[30.03.06 16:28:06]
Отправлено 69 записей		[30.03.06 16:27:58]
Отправлено 99 записей		[30.03.06 16:27:58]
Отправлено 99 записей		[30.03.06 16:27:56]
Отправлено 99 записей		[30.03.06 16:27:55]
Отправлено 99 записей		[30.03.06 16:27:55]
Отправлено 10 записей		[30.03.06 16:27:54]
Отправлено 99 записей		[30.03.06 16:27:54]
Соединение с хранилищем "Test_Arc_S" установлено		[30.03.06 16:27:53]
Попытка подсоединения к хранилищу "Test_Arc_S"...		[30.03.06 16:27:52]
Общие сообщения		
Подсоединён клиент #35178872 с компьютера "KL-ALEX"		[30.03.06 16:27:53]
Соединение с WinDecont установлено		[30.03.06 16:27:11]

6.7 Программный доступ к архивам

6.7.1 Введение

В данном файле справки дается описание COM интерфейсов доступа к архивным данным(хранилище) комплекса DeCont, реализованных в библиотеке depArc.dll. Данная библиотека может быть использована в любых языках программирования, поддерживающих технологию COM.

Для облегчения программирования реализованы классы-обертки над COM-объектами: TdepArcStorage, TdepArcPeriodicAnalogViewRead, TdepArcPeriodicCounterViewRead, TdepArcNoPeriodicAnalogViewRead, TdepArcNoPeriodicCounterViewRead, TdepArcEventViewRead, TdepArcLogBookOGViewRead, работа с которыми возможна в IDE Delphi и C++Builder.

6.7.2 Встроенные типы данных

6.7.2.1 Переопределения

6.7.2.1.1 TdepArcAnalog

```
[
  uuid(11C52DEA-EBE9-4579-AE23-CE77AFDA750D),
  version(1.0),
  public
]
typedef float TdepArcAnalog;
```

Представляет собой значение периодического или не периодического аналога.

6.7.2.1.2 TdepArcCount

```
[
  uuid(D5724928-E05E-43DC-909E-F87B504D2FB2),
  version(1.0),
  public
]
typedef int TdepArcCount;
```

Представляет собой значение периодического или не периодического счетчика.

6.7.2.1.3 TdepArcEvent

```
[
  uuid(836AC92A-B2D4-4092-87A7-DBB07B212BE4),
  version(1.0),
  public
]
typedef short TdepArcEvent;
```

Представляет собой значение события (дискрета).

6.7.2.1.4 TdepArcFilling

```
[
  uuid(1989AEBF-765F-4D4B-AA8D-13E602834A75),
  version(1.0),
  public
]
typedef short TdepArcFilling;
```

Представляет собой процент заполнения периодического аналога или счетчика. Может принимать значение от 0 до 100.

6.7.2.1.5 TdepArcQuality

```
[  
    uuid(CDD299C6-2AA7-4B22-8D54-16F733962877),  
    version(1.0),  
    public  
]  
typedef short TdepArcQuality;
```

Представляет собой значение качества в стандарте OPC (для более подробного описания см. "OPC Data Access Custom Interface Specification"). Используется в паре со значением события, не периодического аналога или счетчика.

6.7.2.2 Множества

6.7.2.2.1 TdepArcEnumTypeArc

```
[
  uuid(1D4DC251-ABD7-41EF-A658-CEFC74C02EEA) ,
  version(1.0)
]
typedef enum tagTdepArcEnumTypeArc
{
  taPeriodicAnalog = 1,
  taPeriodicCounter = 2,
  taEvent = 3,
  taNoPeriodicAnalog = 4,
  taNoPeriodicCounter = 5,
  taLogBookOG = 6
} TdepArcEnumTypeArc;
```

Множество используется для описания типов архивов.

taPeriodicAnalog - архив периодических аналогов;
taPeriodicCounter - архив периодических счетчиков;
taEvent - архив событий;
taNoPeriodicAnalog - архив не периодических аналогов;
taNoPeriodicCounter - архив не периодических счетчиков;
taLogBookOG - архив оперативного журнала.

6.7.2.2.2 TdepArcEnumTypePeriod

```
[
  uuid(8312685B-BFF1-445D-BB38-CF843B2B956E) ,
  version(1.0)
]
typedef enum tagTdepArcEnumTypePeriod
{
  tpSecond = 1,
  tpMinute = 2,
  tpHour = 3,
  tpDay = 4,
  tpMonth = 5,
  tpQuartal = 6,
  tpYear = 7,
  tpNone = 0
} TdepArcEnumTypePeriod;
```

Множество используется для описания типов периодов архивов периодических аналогов и счетчиков.

tpSecond - секунда;
tpMinute - минута;
tpHour - час;
tpDay - сутки;
tpMonth - месяц;
tpQuartal - квартал;
tpYear - год;
tpNone - не задан.

6.7.2.2.3 TdepArcEnumTRParams

```
[
  uuid(54CA13CF-5A01-40D2-8AE3-2BE4963D4DAC),
  version(1.0)
]
typedef enum tagTdepArcEnumTRParams
{
  trpRead = 1,
  trpWrite = 2,
  trpReport = 3
} TdepArcEnumTRParams;
```

Множество используется для описания набора псевдонимов параметров транзакций в БД InterBase/FireBird.

trpRead включает в себя следующие параметры настроек:

```
isc_tpb_read
isc_tpb_read_committed
isc_tpb_rec_version
isc_tpb_nowait
```

Рекомендуется использовать для запроса SELECT. ., который читает данные в сетку. При этом следует использовать транзакцию с доступом "только для чтения" с уровнем изоляции READ COMMITTED, чтобы получить самые "свежие" данные из таблицы, как только они будут обновлены/добавлены. При этом обеспечивается чтение всех подтвержденных другими транзакциями записей, причем без конфликтов с параллельно работающими пишущими и читающими транзакциями. Такую транзакцию можно длительное время держать открытой - сервер не нагружается версиями записей.

trpWrite включает в себя следующие параметры настроек:

```
isc_tpb_write
isc_tpb_concurrency
isc_tpb_nowait
```

Для запроса на изменение/добавление данных можно использовать транзакцию с уровнем изоляции concurrency. Запрос на обновление в этом

trpReport включает в себя следующие параметры настроек:

```
isc_tpb_read
isc_tpb_concurrency
isc_tpb_nowait
```

Для запросов, которые применяются для построения отчетов, однозначно нужно использовать транзакцию с режимом доступа "только для чтения" и с уровнем изоляции concurrency. Такая транзакция будет возвращать строго те данные, что существовали на момент ее запуска, - это очень важная особенность для отчетов, которые строятся за несколько проходов по базе данных.

Данный тип используется для параметра в следующих методах [IdepArcStorage.StartTransaction](#), [IdepArcPeriodicArc.StartTransaction](#), [IdepArcNoPeriodicArc.StartTransaction](#).

6.7.2.2.4 TdepArcEnumViewBooks

```
[
  uuid(E1A5535B-6F52-4475-AFD9-D59C20A9BBEA),
  version(1.0)
]
typedef enum tagTdepArcEnumViewBooks
{
  vbNone = 0,
  vbParBook = 1,
  vbPeriodicArc = 2,
  vbNoPeriodicArc = 3,
  vbReadConfigStd = 4,
}
```

```
vbReadConfigExt = 5,  
vbReadConfigExt2 = 6,  
vbReadConfigStg = 7,  
vbEventStates = 8  
} TdepArcEnumViewBooks;
```

Множество используется для выбора справочника хранилища. Используется при вызове метода [IdepArcStorage.User_ViewBooksStorage](#).

vbNone - не задан;
vbParBook - справочник параметров;
vbPeriodicArc - справочник периодичных архивов;
vbNoPeriodicArc - справочник не периодичных архивов;
vbReadConfigStd - конфигурация вычитывания архивов Деконта;
vbReadConfigExt - конфигурация вычитывания архивов внешних устройств;
vbReadConfigExt2 - конфигурация вычитывания внешних архивов Деконта;
vbReadConfigStg - конфигурация вычитывания архивов удаленного хранилища;
vbEventStates - справочник расшифровки состояний дискретов.

6.7.2.3 Структуры

6.7.2.3.1 TdepArcPar

```
[
  uuid(07EBEE31-CEA5-42FA-9BE7-E3B11FBF4A78) ,
  version(1.0)
]
typedef struct tagTdepArcPar
{
  long Id;
  BSTR Name;
  BSTR Code;
  long IdTask;
  long IdEventState;
  double MaxValue;
  double MinValue;
  VARIANT_BOOL DeleteRecord;
} TdepArcPar;
```

Структура представляет собой описание архивируемого параметра в хранилище.

Описание полей структуры:

long Id - уникальный идентификатор записи в таблице параметров;
BSTR Name - имя параметра(уникально в пределах задачи(IdTask));
BSTR Code - код параметра(уникален в пределах задачи(IdTask));
long IdTask - идентификатор задачи(см. [TdepArcTask](#), [IdepArcStorage.ListTasks](#));
long IdEventState - идентификатор таблицы расшифровки состояния (используется для описания состояния дискретов, не периодич);
double MaxValue - максимальное значение параметра для отображения на графике;
double MinValue - минимальное значение параметра для отображения на графике;
VARIANT_BOOL DeleteRecord - пометка на удаление параметра при очистке хранилища;

6.7.2.3.2 TdepArcTask

```
[
  uuid(79A6C0A3-7527-4E1D-9E8D-42A94BC7E674) ,
  version(1.0)
]
typedef struct tagTdepArcTask
{
  long Id;
  BSTR Name;
} TdepArcTask;
```

Структура представляет собой описание задачи(группы параметров).

Описание полей структуры:

long Id - уникальный идентификатор задачи(см. [TdepArcPar](#));
BSTR Name - уникальное название задачи;

6.7.2.3.3 TdepArcAlias

```
[
  uuid(A58A43C7-5A9D-4045-948F-216870BBD629) ,
  version(1.0)
]
```

```
typedef struct tagTdepArcAlias
{
    BSTR Name;
    BSTR ComputerName;
    BSTR DatabaseNamePath;
} TdepArcAlias;
```

Структура представляет собой описание псевдонима хранилища для подключения к нему (см. [IdepArcStorage.AliasName](#) и [IdepArcStorage.ListAliases](#)).

Описание полей структуры:

BSTR Name - имя псевдонима;
BSTR ComputerName - имя компьютера(указывается для удаленного подключения к хранилищу, для локального - пустая строка);
BSTR DatabaseNamePath - путь к файлам хранилища (одинаково указывается как для локального, так и удаленного подключения);

6.7.2.3.4 TdepArcEventState

```
[
    uuid(01A50C0E-D1EE-4F3B-8132-24A423B1DF82),
    version(1.0)
]
typedef struct tagTdepArcEventState
{
    long Id;
    BSTR Name;
    struct tag IdepArcListEventRefStates * ListEventRefStates;
} TdepArcEventState;
```

Структура представляет собой описание таблицы расшифровки состояния параметра для события, не периодического аналога или счетчика (см. [IdepArcStorage.ListEventStates](#) и [TdepArcPar](#)).

Описание полей структуры:

long Id - уникальный идентификатор таблицы расшифровки состояния;
BSTR Name - название таблицы расшифровки;
struct tag IdepArcListEventRefStates * ListEventRefStates - указатель на интерфейс списка состояний для таблицы рас

6.7.2.3.5 TdepArcEventRefState

```
[
    uuid(9FEADF43-A049-4465-ABBA-D5E9B17F4AB6),
    version(1.0)
]
typedef struct tagTdepArcEventRefState
{
    long Id;
    long Value;
    BSTR RefName;
} TdepArcEventRefState;
```

Структура представляет собой описание расшифровки состояния параметра (см. [IdepArcStorage.ListEventStates](#) и [TdepArcEventState](#)).

Описание полей структуры:

long Id - уникальный идентификатор расшифровки состояния параметра;
long Value - значение параметра;
BSTR RefName - строка расшифровки состояния параметра, соответствующая его значению.

6.7.2.3.6 TdepArcViewGroup

```
[
  uuid(BD4D51FF-ECE9-46C7-A7F7-E603EA06588A),
  version(1.0)
]
typedef struct tagTdepArcViewGroup
{
  long Id;
  BSTR Name;
  struct tag IdepArcListViewGroupRefPar * ListRefPar;
} TdepArcViewGroup;
```

Структура представляет собой описание группы параметров просмотра архивных данных (см. [IdepArcStorage.ListViewGroup](#) и [TdepArcViewGroupRefPar](#)).

Описание полей структуры:

long Id - уникальный идентификатор группы просмотра;
BSTR Name - имя группы просмотра;
struct tag IdepArcListViewGroupRefPar * ListRefPar - указатель на интерфейс списка параметров группы просмотра;

6.7.2.3.7 TdepArcViewGroupRefPar

```
[
  uuid(35A047FB-A00F-4E32-803F-00C3DE4133FC),
  version(1.0)
]
typedef struct tagTdepArcViewGroupRefPar
{
  long Id;
  struct tag IdepArcPar * Par;
  long Color;
  long IndexOrder;
} TdepArcViewGroupRefPar;
```

Структура представляет собой описание параметра в группе просмотра архивных данных (см. [IdepArcStorage.ListViewGroup](#), [TdepArcViewGroup](#) и [TdepArcPar](#)).

Описание полей структуры:

long Id - уникальный идентификатор параметра в группе просмотра;
struct tag IdepArcPar * Par - указатель на структуру архивного параметра;
long Color - цвет параметра для графиков;
long IndexOrder - порядковый номер параметра в группе просмотра.

6.7.2.3.8 TdepArcViewGroupRefParRead

```
[
  uuid(9EBB94B2-9244-4D31-BF15-6E42606019DC),
  version(1.0)
]
typedef struct tagTdepArcViewGroupRefParRead
{
  long IdPar;
  long Color;
  long IndexDataColumn;
} TdepArcViewGroupRefParRead;
```

Структура представляет собой описание параметра группы просмотра, полученного при выборке данных из хранилища (см. [IdepArcListViewGroupRefPar](#), [IdepArcViewRead.ListViewGroupRefParRead](#)).

Описание полей структуры:

long IdPar - уникальный идентификатор параметра в хранилище;

long Color - цвет параметра, соответствующий цвету параметра в группе просмотра (см. [TdepArcViewGroupRefPar](#));

long IndexDataColumn - индекс структуры в массиве из [TdepArcPeriodicAnalogDataRecord](#) или [TdepArcPeriodicCounterDataRecord](#) (если в

6.7.2.3.9 TdepArcPeriodicAnalogDataRow

```
[
  uuid(9D3E7E13-A232-483E-B761-9FA65C6A67EF),
  version(1.0)
]
typedef struct tagTdepArcPeriodicAnalogDataRow
{
  DATE DateTime;
  struct tag TdepArcPeriodicAnalogDataRecord * DataRecord;
} TdepArcPeriodicAnalogDataRow;
```

Структура представляет собой описание строки выборки периодических аналогов в определенное время (см. [TdepArcPeriodicAnalogViewRead](#)).

Описание полей структуры:

DATE DateTime - время данных;

struct tag TdepArcPeriodicAnalogDataRecord * DataRecord - указатель на массив структур [TdepArcPeriodicAnalogDataRecord](#)

6.7.2.3.10 TdepArcPeriodicAnalogDataRecord

```
[
  uuid(BE438C95-EEBF-4065-A79F-4D5F06396365),
  version(1.0)
]
typedef struct tagTdepArcPeriodicAnalogDataRecord
{
  TdepArcAnalog Value;
  TdepArcFilling Filling;
  VARIANT_BOOL HasData;
} TdepArcPeriodicAnalogDataRecord;
```

Структура представляет собой описание значения аналога, используемое при выборке данных из хранилища (см. [TdepArcPeriodicAnalogDataRow](#)).

Описание полей структуры:

TdepArcAnalog Value - значение аналога (см. [TdepArcAnalog](#));

TdepArcFilling Filling - процент заполнения (см. [TdepArcFilling](#));

VARIANT_BOOL HasData - указывает, есть ли данные (если TRUE, то можно использовать значения Value и Filling, иначе - нельзя).

6.7.2.3.11 TdepArcPeriodicCounterDataRow

```
[
  uuid(C8BB8804-B46D-491D-A788-680969BA14CB),
  version(1.0)
]
typedef struct tagTdepArcPeriodicCounterDataRow
{
  DATE DateTime;
  struct tag TdepArcPeriodicCounterDataRecord * DataRecord;
} TdepArcPeriodicCounterDataRow;
```

Структура представляет собой описание строки выборки периодических счетчиков в определенное время (см. [TdepArcPeriodicCounterViewRead](#)).

Описание полей структуры:

DATE DateTime - время данных;

struct tag TdepArcPeriodicCounterDataRecord * DataRecord - указатель на массив структур [TdepArcPeriodicCounterDataRecord](#)

6.7.2.3.12 TdepArcPeriodicCounterDataRecord

```
[
  uuid(2A4E1188-A252-42D3-B54C-8F6909ADFA4A) ,
  version(1.0)
]
typedef struct tagTdepArcPeriodicCounterDataRecord
{
  TdepArcCount Value;
  TdepArcFilling Filling;
  VARIANT_BOOL HasData;
} TdepArcPeriodicCounterDataRecord;
```

Структура представляет собой описание значения счетчика, используемое при выборке данных из хранилища (см. [TdepArcPeriodicCounterDataRow](#)).

Описание полей структуры:

TdepArcCount Value - значение счетчика (см. [TdepArcCount](#));

TdepArcFilling Filling - процент заполнения (см. [TdepArcFilling](#));

VARIANT_BOOL HasData - указывает, есть ли данные (если TRUE, то можно использовать значения Value и Filling, иначе - нельзя).

6.7.2.3.13 TdepArcNoPeriodicAnalogDataRow

```
[
  uuid(360CD98B-F932-4AB2-9EB6-D02AFDF23CC6) ,
  version(1.0)
]
typedef struct tagTdepArcNoPeriodicAnalogDataRow
{
  DATE DateTime;
  TdepArcAnalog Value;
  long IdPar;
  TdepArcQuality Quality;
} TdepArcNoPeriodicAnalogDataRow;
```

Структура представляет собой описание архивных данных для не периодического аналога в определенное время (см. [TdepArcNoPeriodicAnalogViewRead](#)).

Описание полей структуры:

DATE DateTime - время данных;

TdepArcAnalog Value - значение аналога (см. [TdepArcAnalog](#));

long IdPar - уникальный идентификатор параметра в хранилище;

TdepArcQuality Quality - значение качества (см. [TdepArcQuality](#)).

6.7.2.3.14 TdepArcNoPeriodicCounterDataRow

```
[
  uuid(8BC7DA50-EE09-402F-AA77-E7AAA46820FA) ,
  version(1.0)
]
typedef struct tagTdepArcNoPeriodicCounterDataRow
{
  DATE DateTime;
  TdepArcCount Value;
  long IdPar;
}
```

```
TdepArcQuality Quality;  
} TdepArcNoPeriodicCounterDataRow;
```

Структура представляет собой описание архивных данных для не периодического счетчика в определенное время (см. [TdepArcNoPeriodicCounterViewRead](#)).

Описание полей структуры:

```
DATE DateTime - время данных;  
TdepArcCount Value - значение счетчика (см. TdepArcCount);  
long IdPar - уникальный идентификатор параметра в хранилище;  
TdepArcQuality Quality - значение качества (см. TdepArcQuality).
```

6.7.2.3.15 TdepArcEventDataRow

```
[  
  uuid(B4805932-727B-4650-A9FB-504F923F1063),  
  version(1.0)  
]  
typedef struct tagTdepArcEventDataRow  
{  
  DATE DateTime;  
  TdepArcEvent Value;  
  long IdPar;  
  TdepArcQuality Quality;  
} TdepArcEventDataRow;
```

Структура представляет собой описание архивных данных для события в определенное время (см. [TdepArcEventViewRead](#)).

Описание полей структуры:

```
DATE DateTime - время данных;  
TdepArcEvent Value - значение события (см. TdepArcEvent);  
long IdPar - уникальный идентификатор параметра в хранилище;  
TdepArcQuality Quality - значение качества (см. TdepArcQuality).
```

6.7.2.3.16 TdepArcLog

```
[  
  uuid(859B559C-4BD0-4E37-A429-EFD4D8B2AB98),  
  version(1.0)  
]  
typedef struct tagTdepArcLog  
{  
  DATE DateTime;  
  long Kind;  
  BSTR Action;  
  BSTR Object;  
  BSTR Disp;  
  BSTR Comment;  
  short Category;  
  BSTR OPCName;  
  VARIANT_BOOL Kvit;  
  BSTR KvitDisp;  
  DATE KvitDateTime;  
  long Id;  
} TdepArcLogA;
```

Структура представляет собой описание

Описание полей структуры:

6.7.2.3.17 TdepArcReadConfigStgPar

```
[
  uuid(F2B434F0-95DF-4457-8431-FDD33D677685),
  version(1.0)
]
typedef struct tagTdepArcReadConfigStgPar
{
  long Id;
  BSTR Code;
} TdepArcReadConfigStgPar;
```

Структура представляет собой описание

Описание полей структуры:

6.7.2.3.18 TdepArcReadConfigStgArc

```
[
  uuid(25325453-7FFA-4372-830D-8150F74A6641),
  version(1.0)
]
typedef struct tagTdepArcReadConfigStgArc
{
  long Id;
  TdepArcEnumTypeArc TypeArc;
  TdepArcEnumTypePeriod TypePeriod;
  long Period;
  struct tag IdepArcListReadConfigStgPar * ListPar;
  BSTR PeriodRead;
} TdepArcReadConfigStgArc;
```

Структура представляет собой описание

Описание полей структуры:

6.7.2.3.19 TdepArcReadConfigStg

```
[
  uuid(75BEB71E-06A0-4DA6-938E-13209727C0D6),
  version(1.0)
]
typedef struct tagTdepArcReadConfigStg
{
  long La;
  BSTR AliasName;
  struct tag IdepArcListReadConfigStgArc * ListArc;
  long Id;
} TdepArcReadConfigStg;
```

Структура представляет собой описание

Описание полей структуры:

6.7.2.3.20 TdepArcReadConfigExt2Par

```
[
  uuid(886C4FB7-CF20-4AA7-8F0E-0396234AAF34),
  version(1.0)
]
```

```
typedef struct tagTdepArcReadConfigExt2Par
{
    long Id;
    long NPar;
} TdepArcReadConfigExt2Par;
```

Структура представляет собой описание

Описание полей структуры:

6.7.2.3.21 TdepArcReadConfigExt2

```
[
    uuid(02E24BB0-B8AE-41A0-87E4-A7961AD97FFB),
    version(1.0)
]
typedef struct tagTdepArcReadConfigExt2
{
    long La;
    TdepArcEnumTypeArc TypeArc;
    long InstN;
    TdepArcEnumTypePeriod TypePeriod;
    long Period;
    BSTR PeriodRead;
    long ServerClass;
    long Id;
    struct tag IdepArcListReadConfigExt2Par * ListPar;
    long NDevice;
    long NArc;
} TdepArcReadConfigExt2;
```

Структура представляет собой описание

Описание полей структуры:

6.7.2.3.22 TdepArcReadConfigExtPar

```
[
    uuid(64596E47-A050-4E36-A3D3-2C13AE82DC20),
    version(1.0)
]
typedef struct tagTdepArcReadConfigExtPar
{
    long Id;
    long NPar;
} TdepArcReadConfigExtPar;
```

Структура представляет собой описание

Описание полей структуры:

6.7.2.3.23 TdepArcReadConfigExt

```
[
    uuid(FCB3ABE4-3689-4213-8678-1A6FA7C698C5),
    version(1.0)
]
typedef struct tagTdepArcReadConfigExt
{
    long La;
    TdepArcEnumTypeArc TypeArc;
```

```
long InstN;  
long NDevice;  
long NArc;  
TdepArcEnumTypePeriod TypePeriod;  
long Period;  
BSTR PeriodRead;  
long ServerClass;  
long Id;  
struct tag IdepArcListReadConfigExtPar * ListPar;  
} TdepArcReadConfigExt;
```

Структура представляет собой описание

Описание полей структуры:

6.7.2.3.24 TdepArcReadConfigStd

```
[  
    uuid(7EBAABA8-90AD-448F-997B-75205942EEE3),  
    version(1.0)  
]  
typedef struct tagTdepArcReadConfigStd  
{  
    long Id;  
    long La;  
    TdepArcEnumTypeArc EnumTypeArc;  
    long InstN;  
    long NArc;  
    BSTR PeriodRead;  
} TdepArcReadConfigStd;
```

Структура представляет собой описание

Описание полей структуры:

6.7.3 Классы

6.7.3.1 depArcCustomStorage

```
[
  uuid(82EA56EE-D8A2-40D2-8050-8266822DC56C),
  version(1.0),
  helpstring("depArcStorage Object")
]
coclass depArcCustomStorage
{
  [default] interface IdepArcStorage;
  [default, source] dispinterface IdepArcStorageEvents;
};
```

6.7.3.2 depArcCustomPeriodicAnalogViewRead

```
[
  uuid(41D3698B-B280-4373-AA70-AA75C7C3C51C),
  version(1.0),
  helpstring("depArcPeriodicAnalogViewRead Object")
]
coclass depArcCustomPeriodicAnalogViewRead
{
  [default] interface IdepArcPeriodicAnalogViewRead;
  [default, source] dispinterface IdepArcViewReadEvents;
  interface IdepArcViewRead;
};
```

Объект служит для выборки аналоговых данных из хранилища. Перед вызовом метода [ReadBetweenDateTime](#) требуется задать хранилище (свойство `Storage`), тип периода архива (свойство `TypePeriod`), период архива (свойство `Period`), группу просмотра (свойство `ViewGroup`). Результатом выборки будет набор структур [TdepArcPeriodicAnalogDataRow](#), количество таких структур можно узнать из [свойства Count](#).

6.7.3.3 depArcCustomPeriodicCounterViewRead

```
[
  uuid(2C33F795-E137-4B40-91A1-DCAD8DFE766A),
  version(1.0),
  helpstring("depArcPeriodicCounterViewRead Object")
]
coclass depArcCustomPeriodicCounterViewRead
{
  [default] interface IdepArcPeriodicCounterViewRead;
  interface IdepArcViewRead;
  [default, source] dispinterface IdepArcViewReadEvents;
};
```

6.7.3.4 depArcCustomNoPeriodicAnalogViewRead

```
[
  uuid(F4914DB1-9823-48A7-921A-60E2F6B6E6C1),
  version(1.0),
  helpstring("depArcNoPeriodicAnalogViewRead Object")
]
coclass depArcCustomNoPeriodicAnalogViewRead
{
  [default] interface IdepArcNoPeriodicAnalogViewRead;
  interface IdepArcViewRead;
  [default, source] dispinterface IdepArcViewReadEvents;
};
```

6.7.3.5 depArcCustomNoPeriodicCounterViewRead

```
[
  uuid(25C989EE-C184-4C9F-A976-17D067DD0C67),
  version(1.0),
  helpstring("depArcNoPeriodicCounterViewRead Object")
]
coclass depArcCustomNoPeriodicCounterViewRead
{
  [default] interface IdepArcNoPeriodicCounterViewRead;
  interface IdepArcViewRead;
  [default, source] dispinterface IdepArcViewReadEvents;
};
```

6.7.3.6 depArcCustomEventViewRead

```
[
  uuid(1DE700A5-64EF-42DE-B4EC-903C621D2905),
  version(1.0),
  helpstring("depArcEventViewRead Object")
]
coclass depArcCustomEventViewRead
{
  [default] interface IdepArcEventViewRead;
  interface IdepArcViewRead;
  [default, source] dispinterface IdepArcViewReadEvents;
};
```

6.7.3.7 depArcCustomLogBookOGViewRead

```
[
  uuid(4F374EF9-78DD-47F0-B8E8-08179F5B88AF),
  version(1.0),
  helpstring("depArcLogBookOGViewRead Object")
]
coclass depArcCustomLogBookOGViewRead
{
  [default] interface IdepArcLogBookOGViewRead;
  [default, source] dispinterface IdepArcViewReadEvents;
};
```

6.7.4 Интерфейсы

6.7.4.1 IdepArcStorage

```
[
  uuid(8EBDD882-8827-434B-84B7-6F332A1A7A3C),
  version(1.0),
  helpstring("Dispatch interface for depArcStorage Object"),
  dual,
  oleautomation
]
interface IdepArcStorage: IDispatch
{
  [
    id(0x00000001)
  ]
  HRESULT _stdcall OnStartPage([in] IUnknown * AScriptingContext );
  [
    id(0x00000002)
  ]
  HRESULT _stdcall OnEndPage( void );
  [
    id(0x00000003)
  ]
  HRESULT _stdcall User_CreateNewStorage([out] TdepArcAlias ** Alias);
  [
    id(0x00000004)
  ]
  HRESULT _stdcall StartTransaction([in] TdepArcEnumTRParams EnumTRParams);
  [
    id(0x00000005)
  ]
  HRESULT _stdcall Commit( void );
  [
    id(0x00000006)
  ]
  HRESULT _stdcall Rollback( void );
  [
    propget,
    id(0x00000009)
  ]
  HRESULT _stdcall AliasName([out, retval] BSTR * Value );
  [
    propput,
    id(0x00000009)
  ]
  HRESULT _stdcall AliasName([in] BSTR Value );
  [
    id(0x00000007)
  ]
  HRESULT _stdcall Open([in] BSTR DatabaseNamePath, [in] BSTR UserName, [in] BSTR Password );
  [
    id(0x00000008)
  ]
  HRESULT _stdcall Close( void );
  [
    propget,
    id(0x0000000B)
  ]

```

```
]
HRESULT _stdcall Connected([out, retval] VARIANT_BOOL * Value );
[
propget,
id(0x0000000B)
]
HRESULT _stdcall Connected([in] VARIANT_BOOL Value );
[
propget,
id(0x0000000D)
]
HRESULT _stdcall ListPar([out, retval] IdepArcListPar ** Value);
[
propget,
id(0x0000000E)
]
HRESULT _stdcall VCLObject([out, retval] long * Value );
[
propget,
id(0x0000000F)
]
HRESULT _stdcall ListNoPeriodicArc([out, retval] IdepArcListNoPeriodicArc ** Value);
[
propget,
id(0x0000000C)
]
HRESULT _stdcall MainWindowHandle([out, retval] long * Value );
[
propget,
id(0x0000000C)
]
HRESULT _stdcall MainWindowHandle([in] long Value );
[
propget,
id(0x0000000A)
]
HRESULT _stdcall ListEventStates([out, retval] IdepArcListEventStates ** Value);
[
propget,
id(0x00000010)
]
HRESULT _stdcall ListTasks([out, retval] IdepArcListTasks ** Value);
[
propget,
id(0x00000011)
]
HRESULT _stdcall ListReadConfigStd([out, retval] IdepArcListReadConfigStd ** Value);
[
propget,
id(0x00000012)
]
HRESULT _stdcall ListReadConfigExt([out, retval] IdepArcListReadConfigExt ** Value);
[
propget,
id(0x00000013)
]
HRESULT _stdcall ListPeriodicArc([out, retval] depArcListPeriodicArc ** Value);
[
propget,
id(0x00000014)
]
]
```

```
HRESULT _stdcall ListAliases([out, retval] IdepArcListAliases ** Value);
[
id(0x00000015)
]
HRESULT _stdcall StartWriteData([in] BSTR GuidProgram );
[
id(0x00000016)
]
HRESULT _stdcall StopWriteData([in] BSTR GuidProgram );
[
id(0x00000017)
]
HRESULT _stdcall User_ViewBooksStorage([in] TdepArcEnumViewBooks ViewBook);
[
id(0x00000018)
]
HRESULT _stdcall User_AddNewUser( void );
[
propget,
id(0x00000019)
]
HRESULT _stdcall ListReadConfigExt2([out, retval] IdepArcListReadConfigExt2 ** Value);
[
propget,
id(0x0000001A)
]
HRESULT _stdcall ListReadConfigStg([out, retval] IdepArcListReadConfigStg ** Value);
[
propget,
id(0x0000001B)
]
HRESULT _stdcall AutoRelodObjects([out, retval] VARIANT_BOOL * Value );
[
propget,
id(0x0000001B)
]
HRESULT _stdcall AutoRelodObjects([in] VARIANT_BOOL Value );
[
id(0x0000001C)
]
HRESULT _stdcall GetStringErrorOverHRESULTCode([in] HRESULT HRESULT, [out] BSTR * StrError );
[
propget,
id(0x0000001E)
]
HRESULT _stdcall ListViewGroup([out, retval] IdepArcListViewGroup ** Value);
[
id(0x0000001D)
]
HRESULT _stdcall User_ShowListAliases([in] VARIANT_BOOL Connect, [in] VARIANT_BOOL ReadOnly, [out] Tdep
[
id(0x0000001F)
]
HRESULT _stdcall DateTimeToShiftDateTime([in] DATE DateTime, [out] DATE * ShiftDateTime );
[
id(0x00000020)
]
HRESULT _stdcall ShiftDateTimeToDateTime([in] DATE ShiftDateTime, [out] DATE * DateTime );
};
```

6.7.4.1.1 Методы

6.7.4.1.1.1 Rollback

```
[
id(0x00000006)
]
HRESULT _stdcall Rollback( void );
```

6.7.4.1.1.2 Commit

```
[
id(0x00000005)
]
HRESULT _stdcall Commit( void );
```

6.7.4.1.1.3 StartTransaction

```
[
id(0x00000004)
]
HRESULT _stdcall StartTransaction([in] TdepArcEnumTRParams EnumTRParams);
```

При использовании данной библиотеки в основном нет необходимости в использовании транзакций, когда нужно оно запускаются автоматически.

6.7.4.1.1.4 User_CreateNewStorage

```
[
id(0x00000003)
]
HRESULT _stdcall User_CreateNewStorage([out] TdepArcAlias ** Alias);
```

6.7.4.1.1.5 Open

```
[
id(0x00000007)
]
HRESULT _stdcall Open([in] BSTR DatabaseNamePath, [in] BSTR UserName, [in] BSTR Password );
```

Установление соединения с хранилищем, находящимся в DatabaseNamePath. DatabaseNamePath представляет из себя полный путь к хранилищу в формате [Имя компьютера:]<Локальный путь к папке>. Если имя компьютера не указано устанавливается локальное соединение с хранилищем. В UserName передается имя пользователя, в Password - его пароль.

6.7.4.1.1.6 StopWriteData

```
[
id(0x00000016)
]
HRESULT _stdcall StopWriteData([in] BSTR GuidProgram );
```

6.7.4.1.1.7 User_ViewBooksStorage

```
[
id(0x00000017)
]
HRESULT _stdcall User_ViewBooksStorage([in] TdepArcEnumViewBooks ViewBook);
```

6.7.4.1.1.8 User_AddNewUser

```
[
id(0x00000018)
]
HRESULT _stdcall User_AddNewUser( void );
```

6.7.4.1.1.9 GetStringErrorOverHRESULTCode

```
[
id(0x0000001C)
]
HRESULT _stdcall GetStringErrorOverHRESULTCode([in] HRESULT HRESULT, [out] BSTR * StrError );
```

6.7.4.1.1.10 User_ShowListAliases

```
[
id(0x0000001D)
]
HRESULT _stdcall User_ShowListAliases([in] VARIANT_BOOL Connect, [in] VARIANT_BOOL ReadOnly, [out] TdepAr
```

6.7.4.1.1.11 DateTimeToShiftDateTime

```
[
id(0x0000001F)
]
HRESULT _stdcall DateTimeToShiftDateTime([in] DATE DateTime, [out] DATE * ShiftDateTime );
```

6.7.4.1.1.12 ShiftDateTimeToDateTime

```
[
id(0x00000020)
]
HRESULT _stdcall ShiftDateTimeToDateTime([in] DATE ShiftDateTime, [out] DATE * DateTime );
```

6.7.4.1.1.13 StartWriteData

```
[
id(0x00000015)
]
HRESULT _stdcall StartWriteData([in] BSTR GuidProgram );
```

6.7.4.1.1.14 Close

```
[
id(0x00000008)
]
HRESULT _stdcall Close( void );
```

Закрытие соединения с хранилищем.

6.7.4.1.2 Свойства

6.7.4.1.2.1 Connected

```
[
propget,
id(0x0000000B)
]
HRESULT _stdcall Connected([out, retval] VARIANT_BOOL * Value );
```

```
[
propput,
id(0x0000000B)
]
HRESULT _stdcall Connected([in] VARIANT_BOOL Value );
```

6.7.4.1.2.2 AliasName

```
[
propget,
id(0x00000009)
]
HRESULT _stdcall AliasName([out, retval] BSTR * Value );
```

```
[
propput,
id(0x00000009)
]
HRESULT _stdcall AliasName([in] BSTR Value );
```

6.7.4.1.2.3 ListPar

```
[
propget,
id(0x0000000D)
]
HRESULT _stdcall ListPar([out, retval] IdepArcListPar ** Value);
```

Свойство возвращает указатель на интерфейс для взаимодействия со справочником параметров. См. [IdepArcListPar](#).

6.7.4.1.2.4 ListPeriodicArc

```
[
propget,
id(0x00000013)
]
HRESULT _stdcall ListPeriodicArc([out, retval] depArcListPeriodicArc ** Value);
```

Свойство возвращает указатель на интерфейс для взаимодействия со справочником периодичных архивов. См. [IListPeriodicArc](#).

6.7.4.1.2.5 MainWindowHandle

```
[
propget,
id(0x0000000C)
]
HRESULT _stdcall MainWindowHandle([out, retval] long * Value );

[
propget,
id(0x0000000C)
]
HRESULT _stdcall MainWindowHandle([in] long Value );
```

6.7.4.1.2.6 ListNoPeriodicArc

```
[
propget,
id(0x0000000F)
]
HRESULT _stdcall ListNoPeriodicArc([out, retval] IdepArcListNoPeriodicArc ** Value);
```

Свойство возвращает указатель на интерфейс для взаимодействия со справочником не периодичных архивов. См. [IListNoPeriodicArc](#).

6.7.4.1.2.7 ListEventStates

```
[
propget,
id(0x0000000A)
]
HRESULT _stdcall ListEventStates([out, retval] IdepArcListEventStates ** Value);
```

6.7.4.1.2.8 ListTasks

```
[
propget,
id(0x00000010)
]
HRESULT _stdcall ListTasks([out, retval] IdepArcListTasks ** Value);
```

6.7.4.1.2.9 ListReadConfigStd

```
[
propget,
id(0x00000011)
]
HRESULT _stdcall ListReadConfigStd([out, retval] IdepArcListReadConfigStd ** Value);
```

6.7.4.1.2.10 ListReadConfigExt

```
[
propget,
id(0x00000012)
]
HRESULT _stdcall ListReadConfigExt([out, retval] IdepArcListReadConfigExt ** Value);
```

6.7.4.1.2.11 ListPeriodicArc

```
[
propget,
id(0x00000013)
]
HRESULT _stdcall ListPeriodicArc([out, retval] depArcListPeriodicArc ** Value);
```

6.7.4.1.2.12 ListAliases

```
[
propget,
id(0x00000014)
]
HRESULT _stdcall ListAliases([out, retval] IdepArcListAliases ** Value);
```

6.7.4.1.2.13 ListReadConfigExt2

```
[
propget,
id(0x00000019)
]
HRESULT _stdcall ListReadConfigExt2([out, retval] IdepArcListReadConfigExt2 ** Value);
```

6.7.4.1.2.14 ListReadConfigStg

```
[
propget,
id(0x0000001A)
]
HRESULT _stdcall ListReadConfigStg([out, retval] IdepArcListReadConfigStg ** Value);
```

6.7.4.1.2.15 AutoRelodObjects

```
[
propget,
id(0x0000001B)
]
HRESULT _stdcall AutoRelodObjects([out, retval] VARIANT_BOOL * Value );

[
propput,
id(0x0000001B)
]
HRESULT _stdcall AutoRelodObjects([in] VARIANT_BOOL Value );
```

6.7.4.1.2.16 ListViewGroup

```
[
propget,
id(0x0000001E)
]
HRESULT _stdcall ListViewGroup([out, retval] IdepArcListViewGroup ** Value);
```

Свойство возвращает указатель на интерфейс для взаимодействия со списком групп параметров просмотра архивов. См. [IListPeriodicArc](#).

6.7.4.2 IdepArcStorageEvents

```
[
  uuid(2B2F877D-9312-4DD5-8B88-221F8F9F69AF),
  version(1.0),
  helpstring("Events interface for depArcStorage Object")
]
dispinterface IdepArcStorageEvents
{
  properties:
  methods:
  [
    id(0x00000001)
  ]
  HRESULT OnAfterConnect( void );
  [
    id(0x00000002)
  ]
  HRESULT OnBeforeConnect([out] VARIANT_BOOL * DoConnect );
  [
    id(0x00000003)
  ]
  HRESULT OnAfterDisconnect( void );
  [
    id(0x00000004)
  ]
  HRESULT OnBeforeDisconnect( void );
};
```

6.7.4.2.1 Методы

6.7.4.2.1.1 OnBeforeConnect

```
[
  id(0x00000002)
]
HRESULT OnBeforeConnect([out] VARIANT_BOOL * DoConnect );
```

6.7.4.2.1.2 OnAfterConnect

```
[
  id(0x00000001)
]
HRESULT OnAfterConnect( void );
```

6.7.4.2.1.3 OnBeforeDisconnect

```
[
  id(0x00000004)
]
HRESULT OnBeforeDisconnect( void );
```

6.7.4.2.1.4 OnAfterDisconnect

```
[
  id(0x00000003)
]
```

```
HRESULT OnAfterDisconnect( void );
```

6.7.4.3 IdepArcModifiedEvents

```
[
  uuid(B15A692C-3C39-41BC-9AAB-993382396304),
  version(1.0)
]
dispinterface IdepArcModifiedEvents
{
  properties:
  methods:
  [
    id(0x00000001)
  ]
  HRESULT OnModified( void );
};
```

6.7.4.3.1 OnModified

```
[
  id(0x00000001)
]
HRESULT OnModified( void );
```

6.7.4.4 IdepArcListPar

```
[
  uuid(14AFE43F-0280-4EAF-A7D6-37A8E2B7C675),
  version(1.0),
  helpstring("Dispatch interface for depArcListPar Object"),
  dual,
  oleautomation
]
interface IdepArcListPar: IDispatch
{
  [
    propget,
    id(0x00000001)
  ]
  HRESULT _stdcall Count([out, retval] long * Value );
  [
    propget,
    id(0x00000002)
  ]
  HRESULT _stdcall Items([in] long Index, [out, retval] TdepArcPar ** Value);
  [
    id(0x00000003)
  ]
  HRESULT _stdcall Add2([in] BSTR Name, [in] BSTR Code, [in] long IdTask, [in] long IdEventState, [in] do
  [
    id(0x00000005)
  ]
  HRESULT _stdcall FindByCodeAndIdTask([in] long Code, [in] long IdTask, [out] TdepArcPar ** Par);
  [
    id(0x00000006)
  ]
  HRESULT _stdcall FindById([in] long Id, [out] TdepArcPar ** Par);
```

```
[
id(0x00000008)
]
HRESULT _stdcall Add([in] BSTR Name, [in] BSTR Code, [in] long IdTask, [in] long IdEventState, [in] double
[
id(0x00000007)
]
HRESULT _stdcall MarkDeleteParsTask([in] long IdTask );
[
id(0x00000009)
]
HRESULT _stdcall Add3([in] SAFEARRAY(BSTR) NameArray, [in] SAFEARRAY(BSTR) CodeArray, [in] long IdTask, [
];
```

6.7.4.4.1 Методы

6.7.4.4.1.1 Add2

```
[
id(0x00000003)
]
HRESULT _stdcall Add2([in] BSTR Name, [in] BSTR Code, [in] long IdTask, [in] long IdEventState, [in] double
```

6.7.4.4.1.2 FindByCodeAndIdTask

```
[
id(0x00000005)
]
HRESULT _stdcall FindByCodeAndIdTask([in] long Code, [in] long IdTask, [out] TdepArcPar ** Par);
```

6.7.4.4.1.3 FindById

```
[
id(0x00000006)
]
HRESULT _stdcall FindById([in] long Id, [out] TdepArcPar ** Par);
```

6.7.4.4.1.4 Add

```
[
id(0x00000008)
]
HRESULT _stdcall Add([in] BSTR Name, [in] BSTR Code, [in] long IdTask, [in] long IdEventState, [in] double
```

6.7.4.4.1.5 MarkDeleteParsTask

```
[
id(0x00000007)
]
HRESULT _stdcall MarkDeleteParsTask([in] long IdTask );
```

6.7.4.4.1.6 Add3

```
[
id(0x00000009)
]
HRESULT _stdcall Add3([in] SAFEARRAY(BSTR) NameArray, [in] SAFEARRAY(BSTR) CodeArray, [in] long IdTask, [
```

6.7.4.4.2 Свойства

6.7.4.4.2.1 Count

```
[
propget,
id(0x00000001)
]
HRESULT _stdcall Count([out, retval] long * Value );
```

6.7.4.4.2.2 Items

```
[
propget,
id(0x00000002)
]
HRESULT _stdcall Items([in] long Index, [out, retval] TdepArcPar ** Value);
```

6.7.4.5 IdepArcListPeriodicArc

```
[
  uuid(916A6CE4-F9F6-4D7F-8CC9-1B5A92349F6D),
  version(1.0),
  helpstring("Dispatch interface for depArcListPeriodicArc Object"),
  dual,
  oleautomation
]
interface IdepArcListPeriodicArc: IDispatch
{
  [
  propget,
  id(0x00000001)
  ]
  HRESULT _stdcall Count([out, retval] long * Value );
  [
  propget,
  id(0x00000002)
  ]
  HRESULT _stdcall Items([in] long Index, [out, retval] IdepArcPeriodicArc ** Value);
  [
  id(0x00000003)
  ]
  HRESULT _stdcall Add([in] BSTR Name, [in] TdepArcEnumTypeArc TypeArc, [in] long PeriodStorage, [in] long
  [
  id(0x00000005)
  ]
  ]
  HRESULT _stdcall FindById([in] long Id, [out] IdepArcPeriodicArc ** IPeriodicArc);
  [
  id(0x00000006)
  ]
  ]
  HRESULT _stdcall FindByTypeArcAndPeriod([in] TdepArcEnumTypeArc TypeArc, [in] TdepArcEnumTypePeriod Typ
  [
  id(0x00000007)
  ]
  ]
  HRESULT _stdcall Add2([in] BSTR Name, [in] TdepArcEnumTypeArc TypeArc, [in] long PeriodStorage, [in] long
};
```

6.7.4.5.1 Методы

6.7.4.5.1.1 Add

```
[
id(0x00000003)
]
HRESULT _stdcall Add([in] BSTR Name, [in] TdepArcEnumTypeArc TypeArc, [in] long PeriodStorage, [in] long
```

6.7.4.5.1.2 FindById

```
[
id(0x00000005)
]
HRESULT _stdcall FindById([in] long Id, [out] IdepArcPeriodicArc ** IPeriodicArc);
```

6.7.4.5.1.3 FindByTypeArcAndPeriod

```
[
id(0x00000006)
]
HRESULT _stdcall FindByTypeArcAndPeriod([in] TdepArcEnumTypeArc TypeArc, [in] TdepArcEnumTypePeriod TypeP
```

6.7.4.5.1.4 Add2

```
[
id(0x00000007)
]
HRESULT _stdcall Add2([in] BSTR Name, [in] TdepArcEnumTypeArc TypeArc, [in] long PeriodStorage, [in] long
```

6.7.4.5.2 Свойства

6.7.4.5.2.1 Count

```
[
propget,
id(0x00000001)
]
HRESULT _stdcall Count([out, retval] long * Value );
```

6.7.4.5.2.2 Items

```
[
propget,
id(0x00000002)
]
HRESULT _stdcall Items([in] long Index, [out, retval] IdepArcPeriodicArc ** Value);
```

6.7.4.6 IdepArcPeriodicArc

```
[
uuid(2C40E817-F53F-4A6F-AEAA-918E15EB5C5F),
version(1.0),
helpstring("Dispatch interface for depArcPeriodicArc Object"),
dual,
oleautomation
```

```
]
interface IdepArcPeriodicArc: IDispatch
{
    [
    propget,
    id(0x00000001)
    ]
    HRESULT _stdcall Id([out, retval] long * Value );
    [
    propget,
    id(0x00000002)
    ]
    HRESULT _stdcall Name([out, retval] BSTR * Value );
    [
    propget,
    id(0x00000004)
    ]
    HRESULT _stdcall TypeArc([out, retval] TdepArcEnumTypeArc * Value);
    [
    propget,
    id(0x00000005)
    ]
    HRESULT _stdcall PeriodStorage([out, retval] long * Value );
    [
    propget,
    id(0x00000006)
    ]
    HRESULT _stdcall NumberMissingDay([out, retval] long * Value );
    [
    propget,
    id(0x00000007)
    ]
    HRESULT _stdcall DeleteRecord([out, retval] VARIANT_BOOL * Value );
    [
    propget,
    id(0x00000008)
    ]
    HRESULT _stdcall TypePeriod([out, retval] TdepArcEnumTypePeriod * Value);
    [
    propget,
    id(0x00000009)
    ]
    HRESULT _stdcall Period([out, retval] long * Value );
    [
    propget,
    id(0x0000000A)
    ]
    HRESULT _stdcall SizeDataField([out, retval] long * Value );
    [
    id(0x00000003)
    ]
    HRESULT _stdcall AddAnalog([in] long IdPar, [in] DATE DateTime, [in] float Value, [in] short Filling );
    [
    id(0x0000000C)
    ]
    HRESULT _stdcall AddAnalog2([in] SAFEARRAY(long) IdParArray, [in] SAFEARRAY(DATE) DateTimeArray, [in] S
    [
    id(0x0000000D)
    ]
    HRESULT _stdcall AddCounter([in] long IdPar, [in] DATE DateTime, [in] long Value, [in] short Filling );
    [

```

```
id(0x0000000E)
]
HRESULT _stdcall AddCounter2([in] SAFEARRAY(long) IdParArray, [in] SAFEARRAY(DATE) DateTimeArray, [in]
[
id(0x0000000F)
]
HRESULT _stdcall StartTransaction([in] TdepArcEnumTRParams EnumTRParams);
[
id(0x00000010)
]
HRESULT _stdcall Commit( void );
[
id(0x00000011)
]
HRESULT _stdcall Rollback( void );
};
```

6.7.4.6.1 Свойства

6.7.4.6.1.1 Id

```
[
propget,
id(0x00000001)
]
HRESULT _stdcall Id([out, retval] long * Value );
```

6.7.4.6.1.2 Name

```
[
propget,
id(0x00000002)
]
HRESULT _stdcall Name([out, retval] BSTR * Value );
```

6.7.4.6.1.3 TypeArc

```
[
propget,
id(0x00000004)
]
HRESULT _stdcall TypeArc([out, retval] TdepArcEnumTypeArc * Value);
```

6.7.4.6.1.4 PeriodStorage

```
[
propget,
id(0x00000005)
]
HRESULT _stdcall PeriodStorage([out, retval] long * Value );
```

6.7.4.6.1.5 NumberMissingDay

```
[
propget,
id(0x00000006)
```



```
]
HRESULT _stdcall NumberMissingDay([out, retval] long * Value );
```

6.7.4.6.1.6 DeleteRecord

```
[
propget,
id(0x00000007)
]
HRESULT _stdcall DeleteRecord([out, retval] VARIANT_BOOL * Value );
```

6.7.4.6.1.7 TypePeriod

```
[
propget,
id(0x00000008)
]
HRESULT _stdcall TypePeriod([out, retval] TdepArcEnumTypePeriod * Value);
```

6.7.4.6.1.8 Period

```
[
propget,
id(0x00000009)
]
HRESULT _stdcall Period([out, retval] long * Value );
```

6.7.4.6.1.9 SizeDataField

```
[
propget,
id(0x0000000A)
]
HRESULT _stdcall SizeDataField([out, retval] long * Value );
```

6.7.4.6.2 Методы

6.7.4.6.2.1 AddAnalog

```
[
id(0x00000003)
]
HRESULT _stdcall AddAnalog([in] long IdPar, [in] DATE DateTime, [in] float Value, [in] short Filling );
```

6.7.4.6.2.2 AddAnalog2

```
[
id(0x0000000C)
]
HRESULT _stdcall AddAnalog2([in] SAFEARRAY(long) IdParArray, [in] SAFEARRAY(DATE) DateTimeArray, [in] SAF
```

6.7.4.6.2.3 AddCounter

```
[
id(0x0000000D)
]
HRESULT _stdcall AddCounter([in] long IdPar, [in] DATE DateTime, [in] long Value, [in] short Filling );
```

6.7.4.6.2.4 AddCounter2

```
[
id(0x0000000E)
]
HRESULT _stdcall AddCounter2([in] SAFEARRAY(long) IdParArray, [in] SAFEARRAY(DATE) DateTimeArray, [in] SA
```

6.7.4.6.2.5 StartTransaction

```
[
id(0x0000000F)
]
HRESULT _stdcall StartTransaction([in] TdepArcEnumTRParams EnumTRParams);
```

6.7.4.6.2.6 Commit

```
[
id(0x00000010)
]
HRESULT _stdcall Commit( void );
```

6.7.4.6.2.7 Rollback

```
[
id(0x00000011)
]
HRESULT _stdcall Rollback( void );
```

6.7.4.7 IdepArcListNoPeriodicArc

```
[
  uuid(6C9948C9-66EC-4B2C-80EF-85652CD5BBC3),
  version(1.0),
  helpstring("Dispatch interface for depArcListNoPeriodicArc Object"),
  dual,
  oleautomation
]
interface IdepArcListNoPeriodicArc: IDispatch
{
  [
  propget,
  id(0x00000001)
  ]
  HRESULT _stdcall Count([out, retval] long * Value );
  [
  propget,
  id(0x00000002)
  ]
  HRESULT _stdcall Items([in] long Index, [out, retval] IdepArcNoPeriodicArc ** Value);
  [
  id(0x00000003)
  ]
  HRESULT _stdcall Add([in] BSTR Name, [in] TdepArcEnumTypeArc TypeArc, [in] long PeriodStorage, [in] VAR
  [
  id(0x00000004)
  ]
  HRESULT _stdcall FindByTypeArc([in] TdepArcEnumTypeArc TypeArc, [out] IdepArcNoPeriodicArc ** INoPeriod
```

```
[
id(0x00000005)
]
HRESULT _stdcall FindById([in] long Id, [out] IdepArcNoPeriodicArc ** INoPeriodicArc);
[
id(0x00000006)
]
HRESULT _stdcall Add2([in] BSTR Name, [in] TdepArcEnumTypeArc TypeArc, [in] long PeriodStorage, [in] VA
};
```

6.7.4.7.1 Свойства

6.7.4.7.1.1 Count

```
[
propget,
id(0x00000001)
]
HRESULT _stdcall Count([out, retval] long * Value );
```

6.7.4.7.1.2 Items

```
[
propget,
id(0x00000002)
]
HRESULT _stdcall Items([in] long Index, [out, retval] IdepArcNoPeriodicArc ** Value);
```

6.7.4.7.2 Методы

6.7.4.7.2.1 Add

```
[
id(0x00000003)
]
HRESULT _stdcall Add([in] BSTR Name, [in] TdepArcEnumTypeArc TypeArc, [in] long PeriodStorage, [in] VARIA
```

6.7.4.7.2.2 FindByTypeArc

```
[
id(0x00000004)
]
HRESULT _stdcall FindByTypeArc([in] TdepArcEnumTypeArc TypeArc, [out] IdepArcNoPeriodicArc ** INoPeriodic
```

6.7.4.7.2.3 FindById

```
[
id(0x00000005)
]
HRESULT _stdcall FindById([in] long Id, [out] IdepArcNoPeriodicArc ** INoPeriodicArc);
```

6.7.4.7.2.4 Add2

```
[
id(0x00000006)
]
HRESULT _stdcall Add2([in] BSTR Name, [in] TdepArcEnumTypeArc TypeArc, [in] long PeriodStorage, [in] VARIA
```

6.7.4.8 IdepArcNoPeriodicArc

```
[
  uuid(09FB63BD-4DE7-4134-BA2E-3854424E7460),
  version(1.0),
  helpstring("Dispatch interface for depArcNoPeriodicArc Object"),
  dual,
  oleautomation
]
interface IdepArcNoPeriodicArc: IDispatch
{
  [
  propget,
  id(0x00000001)
  ]
  HRESULT _stdcall Id([out, retval] long * Value );
  [
  propget,
  id(0x00000002)
  ]
  HRESULT _stdcall Name([out, retval] BSTR * Value );
  [
  propget,
  id(0x00000003)
  ]
  HRESULT _stdcall TypeArc([out, retval] TdepArcEnumTypeArc * Value);
  [
  propget,
  id(0x00000004)
  ]
  HRESULT _stdcall PeriodStorage([out, retval] long * Value );
  [
  propget,
  id(0x00000005)
  ]
  HRESULT _stdcall NumberMissingDay([out, retval] short * Value );
  [
  propget,
  id(0x00000006)
  ]
  HRESULT _stdcall DeleteRecord([out, retval] VARIANT_BOOL * Value );
  [
  id(0x00000007)
  ]
  HRESULT _stdcall AddRecordLog([in, out] TdepArcLog * Log);
  [
  id(0x00000008)
  ]
  HRESULT _stdcall KvitRecordLog([in] TdepArcLog * Log);
  [
  id(0x00000009)
  ]
  HRESULT _stdcall AddAnalog([in] long IdPar, [in] DATE DateTime, [in] float Value, [in] short Quality );
  [
  id(0x0000000B)
  ]
  HRESULT _stdcall AddAnalog2([in] SAFEARRAY(long) IdParArray, [in] SAFEARRAY(DATE) DateTimeArray, [in] S
  [
  id(0x0000000C)
```

```
]
HRESULT _stdcall AddCounter([in] long IdPar, [in] DATE DateTime, [in] long Value, [in] short Quality );
[
id(0x0000000D)
]
HRESULT _stdcall AddCounter2([in] SAFEARRAY(long) IdParArray, [in] SAFEARRAY(DATE) DateTimeArray, [in]
[
id(0x0000000E)
]
HRESULT _stdcall AddEvent([in] long IdPar, [in] DATE DateTime, [in] short Value, [in] short Quality );
[
id(0x0000000F)
]
HRESULT _stdcall AddEvent2([in] SAFEARRAY(long) IdParArray, [in] SAFEARRAY(DATE) DateTimeArray, [in] SA
[
id(0x00000010)
]
HRESULT _stdcall StartTransaction([in] TdepArcEnumTRParams EnumTRParams);
[
id(0x00000011)
]
HRESULT _stdcall Commit( void );
[
id(0x00000012)
]
HRESULT _stdcall Rollback( void );
};
```

6.7.4.8.1 Методы

6.7.4.8.1.1 AddRecordLog

```
[
id(0x00000007)
]
HRESULT _stdcall AddRecordLog([in, out] TdepArcLog * Log);
```

6.7.4.8.1.2 KvitRecordLog

```
[
id(0x00000008)
]
HRESULT _stdcall KvitRecordLog([in] TdepArcLog * Log);
```

6.7.4.8.1.3 AddAnalog

```
[
id(0x00000009)
]
HRESULT _stdcall AddAnalog([in] long IdPar, [in] DATE DateTime, [in] float Value, [in] short Quality );
```

6.7.4.8.1.4 AddAnalog2

```
[
id(0x0000000B)
]
HRESULT _stdcall AddAnalog2([in] SAFEARRAY(long) IdParArray, [in] SAFEARRAY(DATE) DateTimeArray, [in] SA
```

6.7.4.8.1.5 AddCounter

```
[
id(0x0000000C)
]
HRESULT _stdcall AddCounter([in] long IdPar, [in] DATE DateTime, [in] long Value, [in] short Quality );
```

6.7.4.8.1.6 AddCounter2

```
[
id(0x0000000D)
]
HRESULT _stdcall AddCounter2([in] SAFEARRAY(long) IdParArray, [in] SAFEARRAY(DATE) DateTimeArray, [in] SAFEARRAY(short) QualityArray);
```

6.7.4.8.1.7 AddEvent

```
[
id(0x0000000E)
]
HRESULT _stdcall AddEvent([in] long IdPar, [in] DATE DateTime, [in] short Value, [in] short Quality );
```

6.7.4.8.1.8 AddEvent2

```
[
id(0x0000000F)
]
HRESULT _stdcall AddEvent2([in] SAFEARRAY(long) IdParArray, [in] SAFEARRAY(DATE) DateTimeArray, [in] SAFEARRAY(short) QualityArray);
```

6.7.4.8.1.9 StartTransaction

```
[
id(0x00000010)
]
HRESULT _stdcall StartTransaction([in] TdepArcEnumTRParams EnumTRParams);
```

6.7.4.8.1.10 Commit

```
[
id(0x00000011)
]
HRESULT _stdcall Commit( void );
```

6.7.4.8.1.11 Rollback

```
[
id(0x00000012)
]
HRESULT _stdcall Rollback( void );
```

6.7.4.8.2 Свойства

6.7.4.8.2.1 Id

```
[
propget,
id(0x00000001)
]
```

```
]
HRESULT _stdcall Id([out, retval] long * Value );
```

6.7.4.8.2.2 Name

```
[
propget,
id(0x00000002)
]
HRESULT _stdcall Name([out, retval] BSTR * Value );
```

6.7.4.8.2.3 TypeArc

```
[
propget,
id(0x00000003)
]
HRESULT _stdcall TypeArc([out, retval] TdepArcEnumTypeArc * Value);
```

6.7.4.8.2.4 PeriodStorage

```
[
propget,
id(0x00000004)
]
HRESULT _stdcall PeriodStorage([out, retval] long * Value );
```

6.7.4.8.2.5 NumberMissingDay

```
[
propget,
id(0x00000005)
]
HRESULT _stdcall NumberMissingDay([out, retval] short * Value );
```

6.7.4.8.2.6 DeleteRecord

```
[
propget,
id(0x00000006)
]
HRESULT _stdcall DeleteRecord([out, retval] VARIANT_BOOL * Value );
```

6.7.4.9 IdepArcListEventStates

```
[
  uuid(E0805F0A-BCD7-4E32-9553-22E572F88C7E),
  version(1.0),
  helpstring("Dispatch interface for depArcListEventStates Object"),
  dual,
  oleautomation
]
interface IdepArcListEventStates: IDispatch
{
  [
  propget,
  id(0x00000002)
  ]

```

```
]
HRESULT _stdcall Count([out, retval] long * Value );
[
propget,
id(0x00000003)
]
HRESULT _stdcall Items([in] long Index, [out, retval] TdepArcEventState ** Value);
[
id(0x00000004)
]
HRESULT _stdcall Add([in] long Id, [in] BSTR Name, [out] TdepArcEventState ** EventState);
[
id(0x00000001)
]
HRESULT _stdcall FindById([in] long Id, [out] TdepArcEventState ** EventState);
[
id(0x00000006)
]
HRESULT _stdcall FindByName([in] BSTR Name, [out] TdepArcEventState ** EventState);
};
```

6.7.4.9.1 Свойства

6.7.4.9.1.1 Count

```
[
propget,
id(0x00000002)
]
HRESULT _stdcall Count([out, retval] long * Value );
```

6.7.4.9.1.2 Items

```
[
propget,
id(0x00000003)
]
HRESULT _stdcall Items([in] long Index, [out, retval] TdepArcEventState ** Value);
```

6.7.4.9.2 Методы

6.7.4.9.2.1 Add

```
[
id(0x00000004)
]
HRESULT _stdcall Add([in] long Id, [in] BSTR Name, [out] TdepArcEventState ** EventState);
```

6.7.4.9.2.2 FindById

```
[
id(0x00000001)
]
HRESULT _stdcall FindById([in] long Id, [out] TdepArcEventState ** EventState);
```

6.7.4.9.2.3 FindByName

```
[
id(0x00000006)
]
HRESULT _stdcall FindByName([in] BSTR Name, [out] TdepArcEventState ** EventState);
```

6.7.4.10 IdepArcListEventRefStates

```
[
  uuid(92645B5A-B648-4D28-8053-84D5182C8150),
  version(1.0),
  helpstring("Dispatch interface for depArcEventRefStates Object"),
  dual,
  oleautomation
]
interface IdepArcListEventRefStates: IDispatch
{
  [
  propget,
  id(0x00000001)
  ]
  HRESULT _stdcall Count([out, retval] long * Value );
  [
  propget,
  id(0x00000002)
  ]
  HRESULT _stdcall Items([in] long Index, [out, retval] TdepArcEventRefState ** Value);
  [
  id(0x00000003)
  ]
  HRESULT _stdcall Add([in] long Value, [in] BSTR RefName );
  [
  id(0x00000004)
  ]
  HRESULT _stdcall Add2([in] long Value, [in] BSTR RefName, [out] TdepArcEventRefState ** IEventRefState)
  [
  id(0x00000006)
  ]
  HRESULT _stdcall FindById([in] long Id, [out] TdepArcEventRefState ** IEventRefState);
  [
  id(0x00000007)
  ]
  HRESULT _stdcall FindByValue([in] long Value, [out] TdepArcEventRefState ** IEventRefState);
  [
  id(0x00000008)
  ]
  HRESULT _stdcall FindByRefName([in] BSTR RefName, [out] TdepArcEventRefState ** IEventRefState);
};
```

6.7.4.10.1 Свойства

6.7.4.10.1.1 Count

```
[
propget,
id(0x00000001)
]
```

```
HRESULT _stdcall Count([out, retval] long * Value );
```

6.7.4.10.1.2 Items

```
[
propget,
id(0x00000002)
]
HRESULT _stdcall Items([in] long Index, [out, retval] TdepArcEventRefState ** Value);
```

6.7.4.10.2 Методы

6.7.4.10.2.1 Add

```
[
id(0x00000003)
]
HRESULT _stdcall Add([in] long Value, [in] BSTR RefName );
```

6.7.4.10.2.2 Add2

```
[
id(0x00000004)
]
HRESULT _stdcall Add2([in] long Value, [in] BSTR RefName, [out] TdepArcEventRefState ** IEventRefState);
```

6.7.4.10.2.3 FindById

```
[
id(0x00000006)
]
HRESULT _stdcall FindById([in] long Id, [out] TdepArcEventRefState ** IEventRefState);
```

6.7.4.10.2.4 FindByValue

```
[
id(0x00000007)
]
HRESULT _stdcall FindByValue([in] long Value, [out] TdepArcEventRefState ** IEventRefState);
```

6.7.4.10.2.5 FindByRefName

```
[
id(0x00000008)
]
HRESULT _stdcall FindByRefName([in] BSTR RefName, [out] TdepArcEventRefState ** IEventRefState);
```

6.7.4.11 IdepArcListViewGroup

```
[
uuid(42F468E1-EDF5-460F-A5A6-017DFF443D76),
version(1.0),
helpstring("Dispatch interface for depArcListViewGroup Object"),
dual,
oleautomation
]
```

```
interface IdepArcListViewGroup: IDispatch
{
    [
        propget,
        id(0x00000001)
    ]
    HRESULT _stdcall Count([out, retval] long * Value );
    [
        propget,
        id(0x00000002)
    ]
    HRESULT _stdcall Items([in] long Index, [out, retval] TdepArcViewGroup ** Value);
    [
        id(0x00000003)
    ]
    HRESULT _stdcall Add([in] BSTR Name, [in] long IdParMain, [in] long IndexOrder, [in] VARIANT_BOOL SaveToD
    [
        id(0x00000004)
    ]
    HRESULT _stdcall Delete([in] long Index );
    [
        id(0x00000005)
    ]
    HRESULT _stdcall Reload( void );
};
```

6.7.4.11.1 Свойства

6.7.4.11.1.1 Count

```
[
propget,
id(0x00000001)
]
HRESULT _stdcall Count([out, retval] long * Value );
```

6.7.4.11.1.2 Items

```
[
propget,
id(0x00000002)
]
HRESULT _stdcall Items([in] long Index, [out, retval] TdepArcViewGroup ** Value);
```

6.7.4.11.2 Методы

6.7.4.11.2.1 Add

```
[
id(0x00000003)
]
HRESULT _stdcall Add([in] BSTR Name, [in] long IdParMain, [in] long IndexOrder, [in] VARIANT_BOOL SaveToD
```

6.7.4.11.2.2 Delete

```
[
id(0x00000004)
]
```

```
HRESULT _stdcall Delete([in] long Index );
```

6.7.4.11.2.3 Reload

```
[  
  id(0x00000005)  
]  
HRESULT _stdcall Reload( void );
```

6.7.4.12 IdepArcListViewGroupRefPar

```
[  
  uuid(493471C0-8546-459B-BE11-B419A5CD2098),  
  version(1.0),  
  helpstring("Dispatch interface for depArcListViewGroupRefPar Object"),  
  dual,  
  oleautomation  
]  
interface IdepArcListViewGroupRefPar: IDispatch  
{  
  [  
    propget,  
    id(0x00000001)  
  ]  
  HRESULT _stdcall Count([out, retval] long * Value );  
  [  
    propget,  
    id(0x00000002)  
  ]  
  HRESULT _stdcall Items([in] long Index, [out, retval] TdepArcViewGroupRefPar ** Value);  
  [  
    id(0x00000003)  
  ]  
  HRESULT _stdcall Add([in] long IdPar, [in] long Color, [in] long IndexOrder );  
};
```

6.7.4.12.1 Свойства

6.7.4.12.1.1 Count

```
[  
  propget,  
  id(0x00000001)  
]  
HRESULT _stdcall Count([out, retval] long * Value );
```

6.7.4.12.1.2 Items

```
[  
  propget,  
  id(0x00000002)  
]  
HRESULT _stdcall Items([in] long Index, [out, retval] TdepArcViewGroupRefPar ** Value);
```

6.7.4.12.2 Методы

6.7.4.12.2.1 Add

```
[
id(0x00000003)
]
HRESULT _stdcall Add([in] long IdPar, [in] long Color, [in] long IndexOrder );
```

6.7.4.13 IdepArcViewRead

```
[
  uuid(74876982-07DA-4D66-829D-7E08DD120728),
  version(1.0),
  dual,
  oleautomation
]
interface IdepArcViewRead: IDispatch
{
  [
  id(0x00000001)
  ]
  HRESULT _stdcall ReadBetweenDateTime([in] DATE BeginDate, [in] DATE EndDate );
  [
  propget,
  id(0x00000002)
  ]
  HRESULT _stdcall Count([out, retval] long * Value );
  [
  propget,
  id(0x00000003)
  ]
  HRESULT _stdcall IsModified([out, retval] VARIANT_BOOL * Value );
  [
  propget,
  id(0x00000004)
  ]
  HRESULT _stdcall Storage([out, retval] IdepArcStorage ** Value);
  [
  propput,
  id(0x00000004)
  ]
  HRESULT _stdcall Storage([in] IdepArcStorage * Value);
  [
  propget,
  id(0x00000005)
  ]
  HRESULT _stdcall NameViewGroup([out, retval] BSTR * Value );
  [
  propput,
  id(0x00000005)
  ]
  HRESULT _stdcall NameViewGroup([in] BSTR Value );
  [
  propget,
  id(0x00000006)
  ]
  ]
}
```

```
HRESULT _stdcall ListViewGroupRefParRead([out, retval] IdepArcListViewGroupRefParRead ** Value);  
};
```

Данный интерфейс включает в себя методы и свойства, используемые в классах выборки данных из хранилища (см. [depArcCustomPeriodicAnalogViewRead](#), [depArcCustomPeriodicCounterViewRead](#), [depArcCustomNoPeriodicAnalogViewRead](#), [depArcCustomNoPeriodicCounterViewRead](#), [depArcCustomEventViewRead](#)).

6.7.4.13.1 Методы

6.7.4.13.1.1 ReadBetweenDateTime

```
[  
id(0x00000001)  
]  
HRESULT _stdcall ReadBetweenDateTime([in] DATE BeginDate, [in] DATE EndDate );
```

Метод запрашивает из хранилища данные с BeginDate по EndDate.

6.7.4.13.2 Свойства

6.7.4.13.2.1 Count

```
[  
propget,  
id(0x00000002)  
]  
HRESULT _stdcall Count([out, retval] long * Value );
```

Свойство возвращает количество строк в выборке данных из хранилища.

6.7.4.13.2.2 IsModified

```
[  
propget,  
id(0x00000003)  
]  
HRESULT _stdcall IsModified([out, retval] VARIANT_BOOL * Value );
```

Свойство возвращает признак изменения данных в хранилище после последней выборки данных.

6.7.4.13.2.3 Storage

```
[  
propget,  
id(0x00000004)  
]  
HRESULT _stdcall Storage([out, retval] IdepArcStorage ** Value);
```

```
[
propget,
id(0x00000004)
]
HRESULT _stdcall Storage([in] IdepArcStorage * Value);
```

Свойство возвращает и устанавливает интерфейс архивного хранилища (см. [IdepArcStorage](#)).

6.7.4.13.2.4 NameViewGroup

```
[
propget,
id(0x00000005)
]
HRESULT _stdcall NameViewGroup([out, retval] BSTR * Value );
```

```
[
propget,
id(0x00000005)
]
HRESULT _stdcall NameViewGroup([in] BSTR Value );
```

Свойство возвращает и устанавливает название группы параметров, которые будут запрошены из хранилища при выборке данных.

6.7.4.13.2.5 ListViewGroupRefParRead

```
[
propget,
id(0x00000006)
]
HRESULT _stdcall ListViewGroupRefParRead([out, retval] IdepArcListViewGroupRefParRead ** Value);
```

Свойство возвращает интерфейс на список параметров группы выборки данных (см. [IdepArcListViewGroupRefParRead](#)).

6.7.4.14 IdepArcListViewGroupRefParRead

```
[
  uuid(2E79BF8B-5377-4F36-A3DD-C9282D9A6AB6),
  version(1.0),
  helpstring("Dispatch interface for depArListViewGroupRefParRead Object"),
  dual,
  oleautomation
]
interface IdepArcListViewGroupRefParRead: IDispatch
{
```

```
[
propget,
id(0x00000001)
]
HRESULT _stdcall Count([out, retval] long * Value );
[
propget,
id(0x00000002)
]
HRESULT _stdcall Items([in] long Index, [out, retval] TdepArcViewGroupRefParRead ** Value);
};
```

6.7.4.14.1 Свойства

6.7.4.14.1.1 Count

```
[
propget,
id(0x00000001)
]
HRESULT _stdcall Count([out, retval] long * Value );
```

Свойство возвращает количество параметров в группе [IdeoArcViewRead.NameViewGroup](#) после выборки данных (см. [ReadBetweenDateTime](#))

6.7.4.14.1.2 Items

```
[
propget,
id(0x00000002)
]
HRESULT _stdcall Items([in] long Index, [out, retval] TdepArcViewGroupRefParRead ** Value);
```

Свойство возвращает указатель на структуру параметра выборки данных, соответствующий параметру в группе просмотра (см. [TdepArcViewGroupRefParRead](#)), имеющий номер по порядку Index (нумерация с 0).

6.7.4.15 IdepArcPeriodicAnalogViewRead

```
[
uuid(395B9B02-F0B9-4987-ACA6-920F59F9412E),
version(1.0),
helpstring("Dispatch interface for depArcPeriodicAnalogViewRead Object"),
dual,
oleautomation
]
```



```
interface IdepArcPeriodicAnalogViewRead: IDispatch
{
    [
        propget,
        id(0x00000003)
    ]
    HRESULT _stdcall Items([in] long Index, [out, retval] TdepArcPeriodicAnalogDataRow ** Value);
    [
        propget,
        id(0x00000007)
    ]
    HRESULT _stdcall TypePeriod([out, retval] TdepArcEnumTypePeriod * Value);
    [
        propput,
        id(0x00000007)
    ]
    HRESULT _stdcall TypePeriod([in] TdepArcEnumTypePeriod Value);
    [
        propget,
        id(0x00000008)
    ]
    HRESULT _stdcall Period([out, retval] long * Value );
    [
        propput,
        id(0x00000008)
    ]
    HRESULT _stdcall Period([in] long Value );
};
```

6.7.4.15.1 Свойства

6.7.4.15.1.1 Items

```
[
    propget,
    id(0x00000003)
]
HRESULT _stdcall Items([in] long Index, [out, retval] TdepArcPeriodicAnalogDataRow ** Value);
```

Свойство возвращает указатель на строку выборки данных (см. [TdepArcPeriodicAnalogDataRow](#)) с индексом по порядку Index (нумерация с 0).

6.7.4.15.1.2 TypePeriod

```
[
    propget,
```

```
id(0x00000007)
]
HRESULT _stdcall TypePeriod([out, retval] TdepArcEnumTypePeriod * Value);
```

```
[
propget,
id(0x00000007)
]
HRESULT _stdcall TypePeriod([in] TdepArcEnumTypePeriod Value);
```

Свойство возвращает и устанавливает тип периода архива, участвующего в выборке данных (см. [TdepArcEnumTypePeriod](#)).

6.7.4.15.1.3 Period

```
[
propget,
id(0x00000008)
]
HRESULT _stdcall Period([out, retval] long * Value );
```

```
[
propget,
id(0x00000008)
]
HRESULT _stdcall Period([in] long Value );
```

Свойство возвращает и устанавливает период архива в единицах [TypePeriod](#), участвующего в выборке данных.

6.7.4.16 IdepArcPeriodicCounterViewRead

```
[
  uuid(86E15417-188E-4B09-8834-14935CBFD418),
  version(1.0),
  helpstring("Dispatch interface for depArcPeriodicCounterViewRead Object"),
  dual,
  oleautomation
]
interface IdepArcPeriodicCounterViewRead: IDispatch
{
  [
  propget,
  id(0x00000002)
  ]
  HRESULT _stdcall Items([in] long Index, [out, retval] TdepArcPeriodicCounterDataRow ** Value);
```

```
[
propget,
id(0x00000007)
]
HRESULT _stdcall TypePeriod([out, retval] TdepArcEnumTypePeriod * Value);
[
propput,
id(0x00000007)
]
HRESULT _stdcall TypePeriod([in] TdepArcEnumTypePeriod Value);
[
propget,
id(0x00000008)
]
HRESULT _stdcall Period([out, retval] long * Value );
[
propput,
id(0x00000008)
]
HRESULT _stdcall Period([in] long Value );
};
```

6.7.4.16.1 Свойства

6.7.4.16.1.1 Items

```
[
propget,
id(0x00000002)
]
HRESULT _stdcall Items([in] long Index, [out, retval] TdepArcPeriodicCounterDataRow ** Value);
```

6.7.4.16.1.2 TypePeriod

```
[
propget,
id(0x00000007)
]
HRESULT _stdcall TypePeriod([out, retval] TdepArcEnumTypePeriod * Value);

[
propput,
id(0x00000007)
]
HRESULT _stdcall TypePeriod([in] TdepArcEnumTypePeriod Value);
```

6.7.4.16.1.3 Period

```
[
propget,
id(0x00000008)
]
HRESULT _stdcall Period([out, retval] long * Value );
```

```
[
propput,
id(0x00000008)
]
HRESULT _stdcall Period([in] long Value );
```

6.7.4.17 IdepArcEventViewRead

```
[
uuid(D1B6C4EC-5937-47A4-9711-E926D72A7A1B),
version(1.0),
helpstring("Dispatch interface for depArcEventViewRead Object"),
dual,
oleautomation
]
interface IdepArcEventViewRead: IDispatch
{
[
propget,
id(0x00000001)
]
HRESULT _stdcall Items([in] long Index, [out, retval] TdepArcEventDataRow ** Value);
};
```

6.7.4.17.1 Свойства

6.7.4.17.1.1 Items

```
[
propget,
id(0x00000001)
]
HRESULT _stdcall Items([in] long Index, [out, retval] TdepArcEventDataRow ** Value);
```

6.7.4.18 IdepArcNoPeriodicAnalogViewRead

```
[
  uuid(82E72F5D-10EF-49BE-B2C0-FA8E0525D02B),
  version(1.0),
  helpstring("Dispatch interface for depArcNoPeriodicAnalogViewRead Object"),
  dual,
  oleautomation
]
interface IdepArcNoPeriodicAnalogViewRead: IDispatch
{
  [
    propget,
    id(0x00000001)
  ]
  HRESULT _stdcall Items([in] long Index, [out, retval] TdepArcNoPeriodicAnalogDataRow ** Value);
};
```

6.7.4.18.1 Свойства

6.7.4.18.1.1 Items

```
[
  propget,
  id(0x00000001)
]
HRESULT _stdcall Items([in] long Index, [out, retval] TdepArcNoPeriodicAnalogDataRow ** Value);
```

6.7.4.19 IdepArcNoPeriodicCounterViewRead

```
[
  uuid(DFF1D8CA-0C49-4D2B-AD53-8C5867E3B38A),
  version(1.0),
  helpstring("Dispatch interface for depArcNoPeriodicCounterViewRead Object"),
  dual,
  oleautomation
]
interface IdepArcNoPeriodicCounterViewRead: IDispatch
{
  [
    propget,
    id(0x00000001)
  ]
  HRESULT _stdcall Items([in] long Index, [out, retval] TdepArcNoPeriodicCounterDataRow ** Value);
};
```

6.7.4.19.1 Свойства

6.7.4.19.1.1 Items

```
[
propget,
id(0x00000001)
]
HRESULT _stdcall Items([in] long Index, [out, retval] TdepArcNoPeriodicCounterDataRow ** Value);
```

6.7.4.20 IdepArcLogBookOGViewRead

```
[
uuid(96D0D379-8A52-4033-BD70-E1F5F700D053),
version(1.0),
helpstring("Dispatch interface for depArcLogBookOGViewRead Object"),
dual,
oleautomation
]
interface IdepArcLogBookOGViewRead: IDispatch
{
[
propget,
id(0x00000001)
]
HRESULT _stdcall Storage([out, retval] IdepArcStorage ** Value);
[
propput,
id(0x00000001)
]
HRESULT _stdcall Storage([in] IdepArcStorage * Value);
[
id(0x00000002)
]
HRESULT _stdcall ChangeBetweenDateTime([in] DATE BeginDate, [in] DATE EndDate );
[
id(0x00000003)
]
HRESULT _stdcall ChangeLastCount([in] long LastCount );
[
id(0x00000004)
]
HRESULT _stdcall ChangeLastDateTime([in] DATE LastDate );
[
propget,
```

```
id(0x00000005)
]
HRESULT _stdcall Items([in] long Index, [out, retval] IdepArcLogA ** Value);
[
propget,
id(0x00000006)
]
HRESULT _stdcall Count([out, retval] long * Value );
[
id(0x00000007)
]
HRESULT _stdcall Clear( void );
[
id(0x00000008)
]
HRESULT _stdcall GetRowOverDateTime([in] DATE DateTime, [out] long * Index, [out] IdepArcLogA ** Log);
[
propget,
id(0x00000009)
]
HRESULT _stdcall MinIntervalUpdate([out, retval] long * Value );
[
propget,
id(0x00000009)
]
HRESULT _stdcall MinIntervalUpdate([in] long Value );
};
```

6.7.4.20.1 Свойства

6.7.4.20.1.1 Items

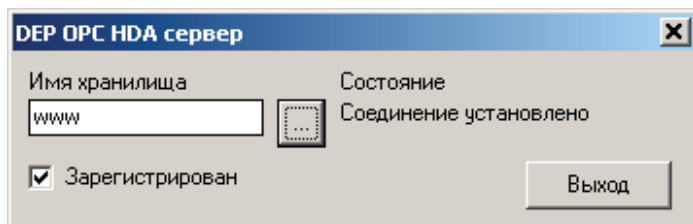
```
[
propget,
id(0x00000001)
]
HRESULT _stdcall Items([in] long Index, [out, retval] TdepArcLogBookOGDataRow ** Value);
```

6.8 OPC HDA Сервер

ПО "SyTrack-WRT" ARC-OPCHDA - OPC HDA сервер архивных данных.

Версия OPC HDA 1.20.

Программа при запуске имеет следующий вид:



В главном окне программы можно изменить имя хранилища, над которым работает программа.

В строке "Состояние" возможны следующие состояния:

- Соединение установлено
- Соединение не установлено

По галочке "Зарегистрирован" можно управлять регистрацией OPC HDA сервера.

При закрытии окна, оно сворачивается в трей.

При перезапуске программы имя хранилища сохраняется.

При именовании архивных элементов, принят следующий шаблон:

Имя_архива\полный_код_параметра.

Имена архивов следующие :

- "аналоги"
- "аналоги по изменению"
- "события"
- "счетчики"
- "счетчики по изменению"

Имя_архива состоит из приведенного имени и периода архивирования, если архив периодический.

Периоды архивирования следующие:

- "сек"
- "мин"
- "часов"
- "дн"
- "мес"
- "кварт"
- "год"