

# **Wonderware<sup>®</sup> FactorySuite<sup>™</sup> IndustrialSQL<sup>™</sup> Server<sup>™</sup>**

## **Руководство администратора системы**

Редакция F  
Июль 1999

**Корпорация Wonderware**

Все права сохранены. Никакая часть настоящей документации не может воспроизводиться, храниться в информационной системе или передаваться любым способом, электронным или механическим, или путем фотокопирования, записи или как-то иначе без предварительного письменного разрешения Wonderware Corporation. Использование содержащейся здесь информации не влечет за собой никакой ответственности, связанной с авторскими или патентными правами. Хотя при подготовке этой документации авторами и издателями были приложены все усилия, они не несут ответственность за возможные ошибки или неточности, равно как за возможный ущерб, причиненный в результате использования содержащейся здесь информации.

Содержащаяся здесь информация может быть изменена без предупреждения и не носит характера обязательств со стороны корпорации Wonderware. Описанное в документации программное обеспечение предоставляется по соглашению о лицензии или неразглашении. Это программное обеспечение может использоваться только на условиях данных соглашений.

**© 1999 Wonderware Corporation. Все права сохранены.**

100 Technology Drive

Irvine, CA 92618

U.S.A.

(949) 727-3200

<http://www.wonderware.com>

#### **Товарные знаки**

Все упоминаемые в этой книге названия, известные как товарные или сервисные знаки, помечены должным образом. Корпорация Wonderware не может гарантировать точность этой информации. Использование любых названий в этой книге не следует считать преднамеренным нарушением каких-либо товарных или сервисных знаков.

Wonderware, InTouch и FactorySuite Web Server являются зарегистрированными товарными знаками Wonderware Corporation.

FactorySuite, Wonderware FactorySuite, WindowMaker, WindowViewer, SQL Access Manager, Recipe Manager, SPCPro, DBDump, DBLoad, HDMerge, HistData, Wonderware Logger, Alarm Logger, InControl, InTrack, InBatch, IndustrialSQL, FactoryOffice, FactoryFocus, License Viewer, Scout, SuiteLink и NetDDE являются зарегистрированными товарными знаками Wonderware Corporation.

## Оглавление

<b>Глава 1 - Введение .....</b>	<b>1-1</b>
Серверные решения для IndustrialSQL .....	1-2
Данные процесса .....	1-2
Традиционные ограничения RDBMS .....	1-3
IndustrialSQL Server как RDBMS реального времени .....	1-4
Архитектура клиент/сервер .....	1-7
Особенности IndustrialSQL Server .....	1-8
Сбор данных .....	1-8
Структуры данных .....	1-8
Хранение данных .....	1-9
Выборка данных .....	1-11
Система управления событиями .....	1-13
Безопасность .....	1-13
Резервное копирование .....	1-14
Мониторинг и диагностика системы .....	1-14
Инструментарий администратора .....	1-15
Инструментарий клиента .....	1-16
Клиентские инструменты Wonderware .....	1-16
Инструменты сторонних разработчиков .....	1-17
<b>Глава 2 - InSQL Configure .....</b>	<b>2-1</b>
О программе InSQL Configure .....	2-2
Запуск программы InSQL Configure .....	2-3
Как ориентироваться в главном окне .....	2-5
Панели инструментов InSQL .....	2-6
Строка состояния .....	2-7
Броузер .....	2-7
Панель свойств .....	2-12
Как показать или скрыть программные элементы .....	2-12
Просмотр информации о программе .....	2-13
Настройка информации о сервере .....	2-14
Просмотр общих свойств сервера .....	2-14
Просмотр архивных блоков .....	2-15
Просмотр информации о сводках .....	2-16
Редактирование именных системных параметров .....	2-19
Настройка информации тэгов .....	2-21
Редактирование конфигурации сбора данных для тэга .....	2-21
Настройка аналоговых тэгов .....	2-24
Настройка единиц измерения .....	2-30
Настройка дискретных тэгов .....	2-33
Настройка пар сообщений .....	2-36
Настройка текстовых тэгов .....	2-39
Настройка тэгов событий .....	2-42
Настройка информации о местах хранения .....	2-42
Настройка серверов ввода-вывода .....	2-43
Повторная инициализация .....	2-44
Хранимые процедуры для повторной инициализации .....	2-45
Отслеживание изменений .....	2-46
Таблицы отслеживания изменений .....	2-46
Особенности и ограничения отслеживания изменений .....	2-46
Включение и выключение отслеживания изменений .....	2-47

Управление дисковым пространством .....	2-48
<b>Глава 3 - Интеграция InTouch .....</b>	<b>3-1</b>
Импортирование словаря тэгов InTouch .....	3-2
Об импортировании .....	3-2
Резервирование базы данных Runtime .....	3-5
Импортирование из InTouch .....	3-6
Повторная инициализация системы .....	3-13
Повторное импортирование .....	3-13
IndustrialSQL как поставщик архива для InTouch .....	3-14
Элементы ActiveX системы IndustrialSQL Server .....	3-14
Использование InTouch SQL Access Manager .....	3-15
Создание соединения с базой данных .....	3-16
Использование сложных запросов .....	3-17
<b>Глава 4 - Управление системой хранения.....</b>	<b>4-1</b>
О системе хранения .....	4-2
Циклическое хранение .....	4-2
Дельта-хранение .....	4-2
Активный образ .....	4-3
Хранение данных конфигурации .....	4-5
Просмотр файлов базы данных .....	4-5
Хранение архивных данных .....	4-7
Расположение архивного блока .....	4-8
Изменение мест хранения .....	4-8
Нотация архивного блока .....	4-9
Просмотр архивных блоков в InSQL Configure .....	4-10
Области хранения архивных блоков .....	4-11
Требования к размеру диска .....	4-13
Требования циклического хранения аналоговых данных .....	4-13
Требования дельта-хранения дискретных данных .....	4-13
Требования дельта-хранения аналоговых данных .....	4-14
Системная перезапись архивных блоков .....	4-14
Запуск системы хранения .....	4-17
Операции системы хранения .....	4-17
Службы IndustrialSQL Server .....	4-19
Резервирование архивных данных .....	4-19
Поддержка перевода часов на летнее/зимнее время .....	4-20
Перевод времени на час вперед .....	4-20
Перевод времени на час назад .....	4-21
Настройка дельта-хранения для быстро изменяющихся тэгов .....	4-23
Как работает дельта-хранение .....	4-23
Дельта-хранение и быстро изменяющиеся тэги .....	4-24
Изменение размера снимков для быстро изменяющихся тэгов .....	4-24
<b>Глава 5 - InSQL Control .....</b>	<b>5-1</b>
О программе InSQL Control .....	5-2
Действие лицензии при запуске системы .....	5-2
Запуск InSQL Control .....	5-3
Строка заголовка .....	5-3
Панель инструментов .....	5-4
Строка состояния системы .....	5-4
Область системных ошибок .....	5-5
Окно просмотра .....	5-5
Строка состояния .....	5-5
Отображение или скрытие программных элементов .....	5-6
Запуск системы .....	5-7
Выполнение процедуры запуска .....	5-7

Системные процессы .....	5-8
Настройка автоматического запуска .....	5-9
Остановка системы .....	5-10
Отслеживание сбора данных .....	5-11
Отслеживание общего счетчика пакетов .....	5-11
Отслеживание сбора данных .....	5-12
Отслеживание деталей сбора данных .....	5-13
Изменение режима просмотра .....	5-14
Сброс счетчика ошибок .....	5-16
Создание нового архивного блока .....	5-16
Изменение учетной записи для службы .....	5-17
Изменение учетной записи для SQL Server .....	5-18

## **Глава 6 - Выборка данных ..... 6-1**

Типы выборки данных .....	6-2
Циклическая выборка .....	6-2
Дельта-выборка .....	6-2
Выборка через службы открытых данных (ODS) .....	6-2
Архитектура ODS IndustrialSQL Server .....	6-3
Приложения-клиенты, использующие ODS .....	6-3
Использование расширений временного домена IndustrialSQL через ODS .....	6-3
Выборка через поставщика данных OLE DB .....	6-4
Нелокальные хранилища данных .....	6-4
Поставщики данных OLE DB .....	6-4
Удаленные таблицы OLE DB .....	6-5
Архитектура поставщика данных OLE DB INSQL .....	6-5
Подготовка к использованию OLE DB .....	6-6
Клиенты IndustrialSQL, использующие OLE DB .....	6-6
Использование расширений временного домена IndustrialSQL через OLE DB .....	6-6
Связь поставщика данных OLE DB INSQL с Microsoft SQL Server .....	6-7
Сводка ODS и OLE DB .....	6-8

## **Глава 7 - Система управления событиями ... 7-1**

О системе управления событиями .....	7-2
Определение событий: тэги событий .....	7-3
Детекторы событий .....	7-4
Действия над событием .....	7-5
Выделение системного потока для событий .....	7-6
Выделение потока для детектора .....	7-6
Выделение потока для действия .....	7-7
Определение края .....	7-8
Архивная запись событий .....	7-9
Компоненты и программы управления событиями .....	7-9
Переменные системы управления событиями .....	7-10
Таблицы хранения событий .....	7-11
Таблицы сводок .....	7-12
Настройка событий .....	7-13
Редактирование общей информации тэга события .....	7-13
Создание тэга события .....	7-16
Выбор внешнего детектора .....	7-19
Настройка общего детектора SQL .....	7-20
Настройка детектора указанного аналогового значения .....	7-22
Настройка детектора указанного дискретного значения .....	7-23
Настройка детектора планировщика .....	7-24
Настройка общего действия SQL .....	7-26
Настройка снимков .....	7-28

Настройка электронного сообщения (E-Mail) .....	7-29
Настройка изменения мертвой зоны .....	7-33
Настройка создания сводок .....	7-34
Поиск тэгов .....	7-39
Удаление тэга события .....	7-42
Настройка безопасности для ActiveEvent .....	7-42
Выборка данных зарегистрированных событий .....	7-47

## **Глава 8 - Мониторинг системы ..... 8-1**

Оценка производительности системы .....	8-2
Отслеживание драйверов системы хранения .....	8-2
Отслеживание переменных целостности системы .....	8-3
Системные переменные .....	8-3
Определение целостности системы .....	8-7
Производительность системы .....	8-8
Счетчики .....	8-8
Режимы просмотра .....	8-9
Загрузка сервера .....	8-10
Журналы событий Windows NT .....	8-11
Просмотр событий Windows NT .....	8-11

## **Глава 9 - Резервное копирование данных .... 9-1**

Копирование и восстановление архивных данных .....	9-2
Копирование архивных блоков .....	9-3
Копирование и восстановление баз данных .....	9-4
Резервное копирование базы данных .....	9-4
Восстановление базы данных .....	9-7

## **Глава 10 - Настройка серверов**

### **ввода-вывода ..... 10-1**

О серверах ввода-вывода .....	10-2
Настройка серверов ввода-вывода через InTouch .....	10-3
Настройка серверов ввода-вывода через InSQL Configure .....	10-4
Редактирование информации сервера ввода-вывода .....	10-4
Добавление сервера ввода-вывода .....	10-7
Удаление сервера ввода-вывода .....	10-9
Редактирование информации тем .....	10-10
Добавление темы .....	10-11
Удаление темы .....	10-12
Синхронизация метки времени и хранение данных .....	10-13
Изменение метки времени на сервере ввода-вывода .....	10-13
Изменение метки времени в IndustrialSQL Server .....	10-14
Работа системы с отключенной синхронизацией .....	10-16
Использование сервера ввода-вывода с несколькими системами IndustrialSQL Server .....	10-17
Сервер ввода-вывода InSQL .....	10-19
Темы сервера ввода-вывода InSQL .....	10-19
Настройки реестра для драйверов устройств .....	10-20
Значение ключей настройки драйверов .....	10-20

## **Глава 11 - IndustrialSQL Server и**

### **Microsoft SQL Server ..... 11-1**

Интеграция с Microsoft SQL Server .....	11-2
Сетевое взаимодействие .....	11-2
Запуск редактора реестра .....	11-3
Обычные соединения с сервером SQL Server .....	11-4
Соединения с IndustrialSQL Server .....	11-5

Настройка IndustrialSQL Server на другие протоколы.....	11-6
Соединения с клиентами.....	11-7
Запуск Microsoft SQL Server.....	11-8
Настройка автоматической загрузки.....	11-9
Использование SQL Server Enterprise Manager.....	11-10
Регистрация сервера.....	11-10
Ориентация в программе SQL Server Enterprise Manager.....	11-11

## **Глава 12 - Соединения с клиентами..... 12-1**

Соединения клиентов.....	12-2
Строка сетевого соединения.....	12-2
Настройки реестра Windows NT.....	12-2
Утилита SQL Server Client Network.....	12-3
Параметры библиотеки БД.....	12-5
Сетевая библиотека.....	12-6
Создание нового псевдонима.....	12-7
Настройка общих параметров клиента.....	12-8
Настройка соединений клиента.....	12-8
Протокол Named Pipes.....	12-9
Протокол NetWare IPX/SPX.....	12-11
Протокол Multi-Protocol.....	12-13
Протокол TCP/IP.....	12-15
Устранение проблем соединения клиентов с IndustrialSQL.....	12-16
Клиенты IndustrialSQL Server 7.0.....	12-16
ODS-клиенты IndustrialSQL Server 7.1.....	12-17
Клиенты OLE DB IndustrialSQL Server 7.1.....	12-18
Правила настройки клиента ODBC.....	12-19

## **Глава 13 - Управление безопасностью..... 13-1**

Уровни безопасности.....	13-2
Уровень 1: безопасность операционной системы.....	13-2
Уровень 2: безопасность Microsoft SQL Server.....	13-2
Уровень 3: безопасность базы данных.....	13-3
Уровень 4: безопасность объектов и выражений.....	13-3
Инструменты безопасности.....	13-5
Проверка режима безопасности.....	13-5
Управление идентификаторами пользователей сервера SQL.....	13-7
Стандартные идентификаторы пользователей IndustrialSQL Server.....	13-7
Добавление идентификатора пользователя.....	13-8
Управление пользователями и группами.....	13-12
Стандартные пользователи и группы IndustrialSQL Server.....	13-12
Добавление пользователя.....	13-14
Добавление новой группы базы данных.....	13-15
Управление правами допуска.....	13-16
Установка прав допуска к объекту.....	13-16
Установка прав доступа к выражениям.....	13-19
Управление паролями.....	13-20
Изменение идентификатора пользователя wwAdmin.....	13-21

## **Глава 14 - Регистрация ошибок..... 14-1**

Управление регистрацией ошибок.....	14-2
Категории ошибок.....	14-2
Настройка места регистрации ошибок.....	14-3
Настройка журнала событий Windows NT.....	14-4
Просмотр зарегистрированной информации IndustrialSQL Server в журнале событий.....	14-5
Расположение файлов регистрации ошибок.....	14-6

Запуск системы и статус подсистемы сервера .....	14-6
Запуск и статус системы хранения .....	14-6
Сообщения об ошибках драйвера ввода/вывода .....	14-7

## **Приложение А - Настройки реестра**

<b>IndustrialSQL Server .....</b>	<b>A-1</b>
Редактирование реестра .....	A-2
Ключи IndustrialSQL Server .....	A-2
Пути хранения данных .....	A-3
Программы .....	A-4
Выборка данных .....	A-7
Установка .....	A-8
Хранение данных .....	A-9
Ключи Microsoft SQL Server .....	A-10
Ключи клиента IndustrialSQL .....	A-11
Ключи FactorySuite .....	A-11
Системные настройки .....	A-12
Регистрация ошибок .....	A-12
Службы .....	A-12

## **Приложение В - Словарь терминов .....**

## **Приложение I - Предметный указатель .....**

## Предисловие

В данном *Руководстве администратора системы IndustrialSQL* описывается, каким образом следует администрировать и сопровождать установленную систему сервера IndustrialSQL™. Данная документация не является руководством по установке ни сервера IndustrialSQL, ни приложений клиента. Как ссылки, так и содержание этого руководства описывают доступный инструментарий системного администратора сервера IndustrialSQL, а также шаги, необходимые для создания архитектуры «клиент-сервер». Здесь также приводится описание системы хранения данных, процедур архивирования данных и регистрации ошибок. Кроме того, имеется краткое руководство по устранению неполадок.

Настоящее руководство предназначено для системных администраторов сервера IndustrialSQL. Сервер IndustrialSQL непосредственно связан с продуктами Microsoft BackOffice; необходимо также знание операционных систем Microsoft® SQL Server™ и Microsoft® Windows NT®. Настоящее руководство предполагает, что вы знакомы с управлением сервером Microsoft SQL и использованием инструментария администратора, поставляемого с операционной системой Microsoft Windows NT.

Более подробная информация о сервере Microsoft SQL и об операционной системе Microsoft Windows NT содержится в документации Microsoft.

## Новые возможности IndustrialSQL Server 7.1

В версии FactorySuite 2000 7.1 сервер IndustrialSQL снабжен новыми расширенными функциями. Дополнительные свойства сервера IndustrialSQL включают поддержку сервера Microsoft SQL 7.0, встроенные и улучшенные системы сводок и управления событиями, поддержку дельта-реимпорта, а также возможность работы в реальном режиме времени с использованием нового сервера ввода-вывода IndustrialSQL.

## Поддержка сервера Microsoft SQL 7.0

В сервере IndustrialSQL сохранены все свойства сервера Microsoft SQL 7.0. Эти новые возможности включают улучшенное управление хранением данных, а также дополнения к приложениям, предназначенным для управления и запросов базы данных. Но наиболее важное новшество сервера Microsoft SQL 7.0 (для сервера IndustrialSQL Server) – улучшенный обработчик запросов.

В более ранних версиях IndustrialSQL Server грамматически проверял все поступающие запросы клиента, передавал запросы на некоторые типы данных в Microsoft SQL Server, а запросы на данные архивных блоков выполнял с помощью собственного механизма выборки данных. Теперь Microsoft SQL Server 7.0 может успешно проверять и обрабатывать все запросы клиента, вне зависимости от того, какой тип данных запрашивается, будь то база данных SQL Server, файлы с данными, базы данных Oracle и т.п. Технология доступа к данным из различных, структурированных или неструктурированных, источников, таких как указанные выше, называется Object Linking and Embedding for Databases или OLE DB (Связывание и встраивание объектов для баз данных).

Использование улучшенного обработчика запросов Microsoft SQL Server делает объединение данных из базы данных Microsoft SQL Server и из архивных блоков гораздо легче и эффективнее. Несколько примеров улучшений:

- Повышенная возможность связывания дискретных и аналоговых таблиц.
- Поддержка курсоров базы данных. Это позволяет проводить достаточно сложные связи, такие как определение средних или накопительных аналоговых значений (например, поток) когда включен дискретный тэг (например, клапан).

Чтобы приложение-клиент получило доступ к данным через OLE DB, запрос должен быть составлен с использованием правильного синтаксиса OLE DB. К инструментам клиента IndustrialSQL, использующим OLE DB для доступа к данным, относятся: InSQL Report, InSQL Configure, а также элементы управления ActiveX IndustrialSQL —ActiveDataGrid, ActiveGraph, ActiveTagBrowser и ActiveTimeSelector.

[См. также главу 6, "Доступ к данным"](#).

## **Интеграция сводок в систему управления событиями**

Система сводок теперь интегрирована в систему управления событиями и улучшена. Сводки все так же хранятся в таблицах сводок, но их выполнение осуществляется системой управления событиями. Система управления событиями теперь включает в себя временной детектор событий, который называется планировщиком. Планировщик определяет, если событие происходит в определенное время или позже по системным часам. Этот новый детектор наиболее полезен для определения времени сводок. Сводки теперь используются в системе управления событиями как особый тип операций и более гибко настраиваются пользователем; можно выбрать любое число тэгов для любого числа расчетов.

[См. также главу 7, "Система управления событиями"](#).

## Улучшения в производительности системы управления событиями

Система управления событиями усовершенствована в плане функциональности и производительности. Поскольку основную нагрузку на систему создают действия, выполняемые при возникновении событий, эти действия теперь пользуются собственным пулом потоков. Это позволяет выполнять несколько действий одновременно, что повышает общую производительность.

К другим улучшениям относятся оптимизация снимочных действий (действий по сохранению значений тэгов в момент возникновения события) и установка приоритетов событий. События с критическим приоритетом обрабатываются в первую очередь, после них обрабатываются события с обычным приоритетом.

См. также главу 7, "Система управления событиями".

## Дельта-реимпортирование

В предыдущих версиях IndustrialSQL Server, если вы импортировали данные темы из приложения InTouch, а затем добавляли или изменяли некоторые тэги в InTouch, необходимо было заново импортировать *все* тэги в IndustrialSQL Server. Если приложения InTouch имели большое количество тэгов, процесс реимпортирования отнимал очень много времени. Теперь IndustrialSQL Server позволяет реимпортировать только те тэги, которые были добавлены или изменены с момента последнего импортирования, что значительно ускоряет процесс.

См. также главу 3, "Интергация с InTouch".

## Сервер ввода-вывода IndustrialSQL Server (InSQLIOS)

Сервер ввода-вывода IndustrialSQL служит в качестве интерфейса для получения клиентами доступа к данным реального времени из IndustrialSQL Server через протокол DDE или SuiteLink. Клиентами могут быть InTouch, FactoryFocus, Visual Basic и т.п.

Сервер ввода-вывода InSQL может использоваться программой WindowViewer для доступа к системным тэгам IndustrialSQL Server, предназначенным для мониторинга состояния системы. WindowViewer можно настроить на выдачу предупреждений при обнаружении ненормальных характеристик системы IndustrialSQL Server. Сервер ввода-вывода IndustrialSQL также предоставляет приложениям, таким как InTouch, доступ к текущим значениям данных (данным реального времени) без необходимости обращаться к SQL.

См. также главу 10, "Настройка серверов ввода/вывода".

## Условные обозначения

В настоящем руководстве используются следующие условные обозначения:

Обозначение	Трактовка
ВЕРХНИЙ РЕГИСТР	Пути, названия файлов и синтаксис Industrial-SQL™.
СМЕШАННЫЙ регистр	Сокращенный синтаксис. Сегмент в верхнем регистре обязателен. Сегмент в нижнем регистре необязателен.
МАЛЫЕ ЗАГЛАВНЫЕ	Названия клавиш, например ESC,
<b>полужирный</b>	Системные процедуры, утилиты, команды и текст, вводимый пользователем.
<i>курсив</i>	Названия баз данных, таблиц, столбцов, индексов и тэгов.
Пропорциональный	Текст на экране, сообщения об ошибках и примеры.
{фигурные скобки}	Обязательные элементы синтаксиса. Сами скобки вводить в синтаксисе не нужно.
[квадратные скобки]	Необязательные элементы синтаксиса. Сами скобки вводить в синтаксисе не нужно.
(вертикальная черта)	Равнозначна "или". В синтаксисе можно ввести только один из двух элементов, разделенных вертикальной чертой.
... (многоточие)	Предшествующий элемент синтаксиса можно повторять.

## Дополнительные источники информации

В дополнение к этому руководству информацию о системе IndustrialSQL Server можно почерпнуть из следующих источников.

### Электронная справка

Электронная справка вызывается из меню справки в любом приложении-клиенте IndustrialSQL Server.

### Электронная документация по IndustrialSQL Server

Электронная документация по IndustrialSQL Server доступна на установочном компакт-диске FactorySuite. Во время установки IndustrialSQL Server вы можете выбрать установку электронной документации на жесткий диск.

Набор электронной документации IndustrialSQL Server включает электронные версии следующих книг по IndustrialSQL Server:

*Руководство администратора системы FactorySuite*

Стандартная книга по FactorySuite, объясняющая установку, настройку, сетевые возможности, интеграцию и стандартные компоненты.

*Руководство по подготовке к работе с IndustrialSQL Server*

Содержит сведения об использовании базовых и расширенных возможностей IndustrialSQL Server и клиентских программ.

*Руководство администратора системы IndustrialSQL Server*

Описывает процедуры сопровождения и администрирования IndustrialSQL Server.

*Руководство по клиентским инструментам IndustrialSQL Server*

Описывает все имеющиеся клиентские инструменты Wonderware, используемые для получения информации из IndustrialSQL Server.

*Справочное руководство IndustrialSQL Server*

Предоставляет подробное описание всех таблиц баз данных IndustrialSQL Server и процедур доступа к информации.

*Руководство по элементам ActiveX IndustrialSQL Server*

Описывает элементы управления ActiveX IndustrialSQL и процедуры их настройки во время разработки приложения.

## Техническая поддержка

Служба технической поддержки корпорации Wonderware располагает обширными возможностями, позволяющими решить множество вопросов по любым аспектам использования продуктов Wonderware. Подробные сведения о службе технической поддержки приводятся в *Руководстве администратора системы FactorySuite*.

## Г Л А В А 1

# Введение

В данной главе рассказывается о преимуществах IndustrialSQL Server, а также приводится краткий обзор возможностей IndustrialSQL Server. В эту главу также включены описания приложений, используемых для управления системой и формирования запросов к ней.

### Содержание

- Серверные решения для IndustrialSQL
- Особенности IndustrialSQL Server
- Инструментарий администратора
- Инструментарий клиента

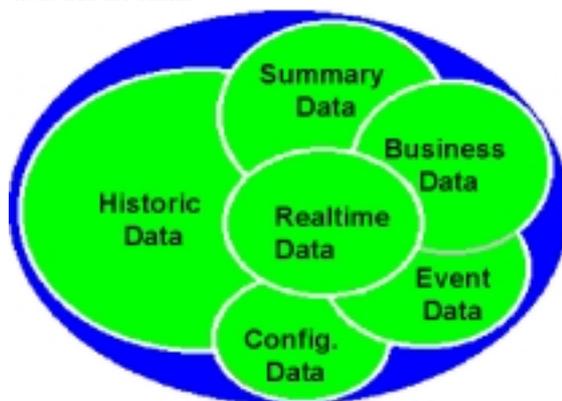
## Серверные решения для IndustrialSQL

IndustrialSQL Server является реляционной системой управления базой данных для хранения производственных данных в реальном режиме времени. IndustrialSQL Server собирает и сохраняет данные производственных процессов с полным разрешением и предоставляет как архивные данные, так и данные реального времени, а также данные конфигурации, событий, сводок и другую производственную информацию в клиентские приложения. IndustrialSQL Server сочетает мощность и гибкость Microsoft SQL Server с высокой скоростью сбора и архивирования данных системы реального времени.

### Данные процесса

Данные процесса представляют собой любой вид информации, связанной с успешным выполнением процесса. Нижеперечисленные виды информации рассматриваются как части данных процесса:

- Данные реального времени - Каково текущее значение данного тэга?
- Архивные данные - Каково было посекундное значение данного тэга в прошлый понедельник?
- Сводные данные - Каково среднее значение для каждого из пяти данных тэгов?
- Бизнес-данные - Какова стоимость отдельного материала?
- Данные событий - Когда котел перегрелся?
- Данные конфигурации - Сколько серверов ввода-вывода используются и каковы их типы?



Для улучшения производительности и качества работы при одновременном снижении расходов вся эта собранная информация должна быть доступной для анализа. Производственные данные обычно подвергаются анализу для следующих целей:

- Диагностика и оптимизация процессов.
- Управление расходом материалов, например, сырья.
- Профилактика оборудования.
- Контроль за качеством продукции и процесса (SPC/SQC).
- Соблюдение норм охраны труда и экологии (EPA/FDA).

- Производственные отчеты.
- Анализ неполадок.

## Традиционные ограничения RDBMS

В таких сферах, как финансы и торговля, задачи исследования интегрированных данных разного типа, полученных из различных источников, успешно решаются при помощи технологии RDBMS (реляционной системы управления базой данных). В технологии RDBMS информация хранится во множестве связанных между собой таблиц. Хранение данных в RDBMS облегчает и ускоряет получение данных.

Язык SQL, используемый для обращения к реляционной базе данных, является промышленным "супер-стандартом", его поддерживают сотни поставщиков программного обеспечения. Он обеспечивает незаменимую в условиях предприятия открытость. На сегодняшний момент реляционные базы данных являются развитыми и общепринятыми «рабочими лошадками» в приложениях баз данных. Своей мощностью и гибкостью SQL значительно превосходит возможности частных решений, которые разрабатывались предприятиями самостоятельно. Кроме того, проблемы интеграции решаются поставщиками баз данных и микропрограммных средств согласованно, что обеспечивает беспрецедентный уровень интеграции в промышленных условиях.

Итак, почему типичную систему RDBMS не используют для хранения производственных данных? Потому что системы RDBMS традиционно имеют следующие ограничения:

- Невозможно справиться с большим объемом данных, генерируемых на производстве.
- Невозможно справиться с высокой скоростью поступления данных.
- SQL неэффективно работает с данными временной последовательности.

На промышленных предприятиях существуют тысячи аналоговых и дискретных тэгов, причем все они изменяются с различной интенсивностью. Результатом нескольких месяцев производственной деятельности являются сотни гигабайт данных в обычной реляционной базе данных.

Помимо этого, предприятие, на котором есть 10 000 переменных, изменяющихся в среднем каждые 2 секунды, создает 5 000 значений в секунду. Следовательно, для того, чтобы хранить весь архив, нужно каждую секунду вводить в базу 5 000 строк данных, а подобная скорость недоступна для обычных реляционных СУБД типа Oracle или SQL Server.

## IndustrialSQL Server как RDBMS реального времени

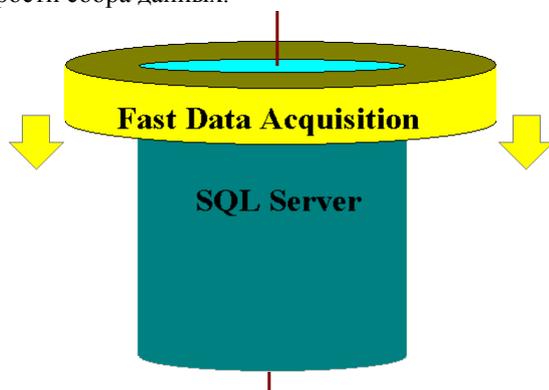
Будучи системой RDBMS реального времени, IndustrialSQL Server является расширением Microsoft SQL Server, повышая более чем на порядок скорость сбора данных, соответственно уменьшая объем сохраняемых данных и предоставляя удобные расширения языка SQL для обработки временных последовательностей.

### Высокоскоростной сбор данных

Широчайший диапазон серверов ввода-вывода корпорации Wonderware позволяет использовать более 500 различных промышленных устройств контроля и сбора данных.

Будучи оптимизирован для быстрого сбора и сохранения аналоговых и дискретных данных, IndustrialSQL Server превосходит все обычные реляционные СУБД, работающие при схожем аппаратном обеспечении, и обладает такой производительностью, которая делает возможным сохранение высокоскоростных данных в реляционной базе данных. В IndustrialSQL Server сбор и сохранение данных производственных процессов осуществляется в 3000 раз быстрее, чем в RDBMS.

Все новые серверы ввода-вывода корпорации Wonderware поддерживают протокол SuiteLink, который поддерживает метки времени и качества данных на сервере ввода-вывода и обеспечивает дополнительное повышение скорости сбора данных.



## Уменьшенные требования дискового пространства

Для хранения данных IndustrialSQL Server занимает такое же пространство, что и обычная реляционная СУБД. Фактическое дисковое пространство, необходимое для хранения производственных данных, зависит от размера и рода деятельности предприятия, а также периода архивного хранения. [Подробнее об требованиях к дисковому пространству см. в главе 4 "Управление системой хранения".](#)

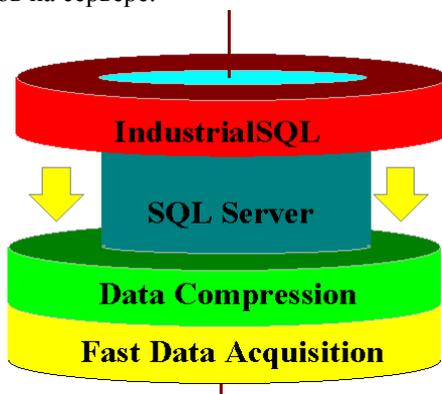
Используемый алгоритм сжатия обеспечивает накопление данных без потерь с высокой степенью разрешения и качества данных.



## Расширения временного домена SQL

Язык SQL не поддерживает некоторые типы данных временной последовательности. В частности, в SQL невозможно управлять разрешением возвращаемых данных и предоставлять клиентам незатребованные данные. Примером функции разрешения является выборка строк данных, равномерно распределенных за определенный период времени.

Сервер Microsoft SQL имеет собственные дополнения к языку SQL, в совокупности известные как Transact-SQL. IndustrialSQL Server расширяет возможности Transact-SQL, позволяя управлять разрешением и незатребованными обновлениями, создавая тем самым основу для работы временных функций, таких как скорость изменения и внутренние расчеты процессов на сервере.

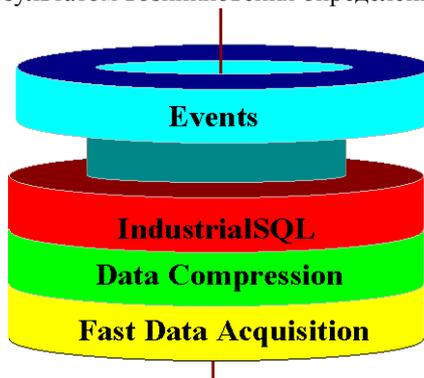


## Система управления событиями

С точки зрения событий, наиболее интересными являются непрерывные данные. Событием может быть все, что угодно — от завершения производства партии изделий до заступления на работу другой смены или запуска производственной линии.

IndustrialSQL Server распознает события (изменения в тэгах, операции вставки, обновления и удаления SQL, либо их комбинации), и инициирует ответные действия. Например, результатом запуска двигателя может быть почтовое сообщение в службу технической поддержки.

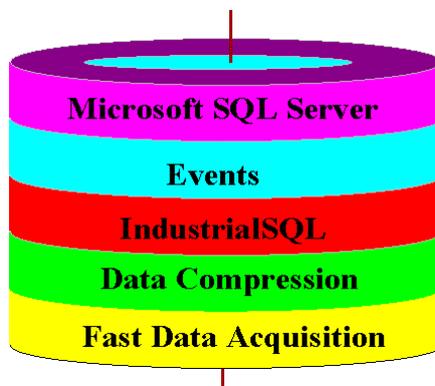
Производственные отчеты содержат, как правило, сводки либо сгруппированные данные. Система управления событиями используется для автоматического обновления таблиц сводок периодическими средними, суммами, а также минимальными и максимальными значениями или агрегациями, являющимися результатом возникновения определенных событий.



## Сервер Microsoft SQL — часть IndustrialSQL Server

Многие производственные данные обладают теми же свойствами, что и обычные бизнес-данные. Например, данные конфигурации или сводок. Данные конфигурации хранятся в таблицах SQL вместе с полным архивом изменений. По мере работы предприятия происходит добавление и удаление тэгов, меняются их определения и технические параметры. Архив изменений обеспечивает согласование данных конфигурации с архивными данными, возвращаемыми сервером. Управление этими данными осуществляется, благодаря нашему длительному сотрудничеству с Microsoft и включению системы Microsoft SQL Server в состав IndustrialSQL Server.

Менеджер SQL Server отвечает за управление учетными записями, группами пользователей (ролями) и правами доступа. Сотни приложений, совместимых с Microsoft SQL Server, совместимы и с IndustrialSQL Server.

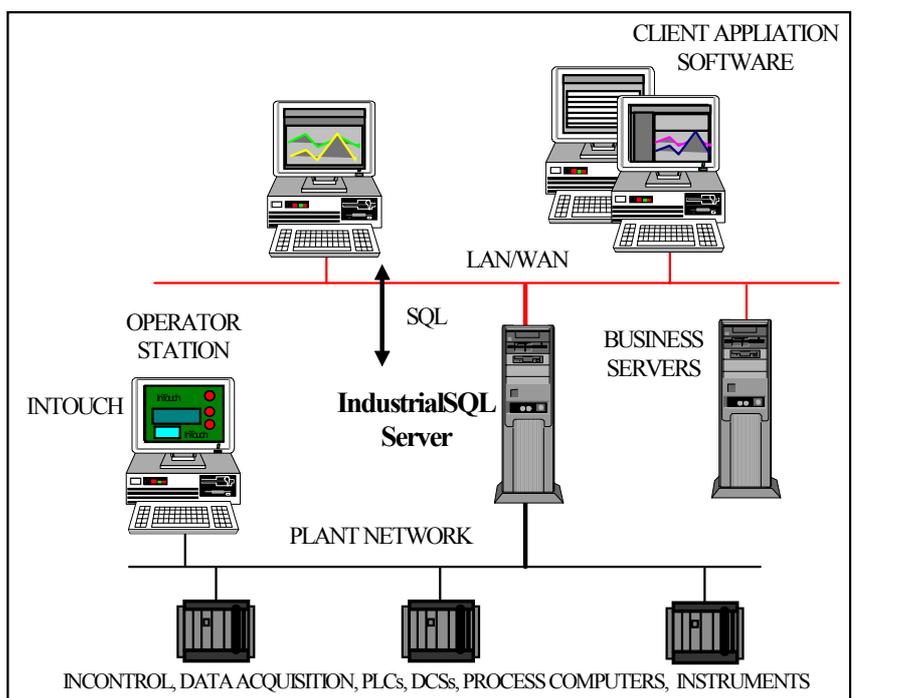


## Архитектура клиент/сервер

Архитектура клиент/сервер IndustrialSQL Server предоставляет пользователям гибкие и удобные приложения, обеспечивая в то же время целостность и безопасность данных на сервере. Подобная архитектура клиент/сервер обеспечивает унифицированный доступ к производственным данным и данным процесса: реального времени и архивным, совокупным данным конфигурации, событий, сводок по бизнес-данным. С целью повышения общей производительности процессорных ресурсов клиента и сервера наиболее интенсивные операции выполняются на сервере и минимизируется передача данных по сети.

Использование в IndustrialSQL Server стандартного интерфейса между серверными службами и клиентскими компьютерами обеспечивает незаменимый в производственной сфере уровень открытости и гибкости программного обеспечения.

С точки зрения предприятия, IndustrialSQL Server является «сердцем» пакета FactorySuite корпорации Wonderware, объединенным с InTouch и получающим данные через серверы ввода-вывода Wonderware из наиболее популярных устройств контроля и сбора данных. На приведенной ниже иллюстрации показан один из возможных вариантов конфигурации сети, где IndustrialSQL Server используется в качестве связующего звена между сетью процесса и управленческой локально-глобальной сетью (LAN/WAN):



К числу клиентов IndustrialSQL Server относятся инструментарий корпорации Wonderware и сотни популярных коммерческих приложений, обеспечивающих мониторинг, анализ, визуализацию и генерацию отчетов для производственных данных.

Более подробно об использовании архитектуры клиент/сервер в пакете FactorySuite см. в *Руководстве администратора системы FactorySuite*.

## Особенности IndustrialSQL Server

IndustrialSQL Server является базовым компонентом системы IndustrialSQL Server, заполняющим разрыв между средой мониторинга и управления производственными данными большого объема в реальном времени и открытой гибкой средой бизнес-управления.

IndustrialSQL Server, включающий в себя систему Microsoft SQL Server, получает данные процессов через высокоскоростные серверы ввода-вывода Wonderware, сжимает и сохраняет их, а затем предоставляет их клиентам в ответ на SQL-запросы. IndustrialSQL Server также хранит информацию о событиях, сводках, конфигурации, безопасности, резервирования и мониторинга системы.

## Сбор данных

IndustrialSQL Server может принимать данные через FastDDE более чем с 500 различных серверов ввода-вывода Wonderware и других производителей, обеспечивающих доступ к наиболее полному набору промышленных устройств контроля и сбора данных. IndustrialSQL Server также поддерживает новое поколение высокопроизводительных серверов ввода-вывода SuiteLink корпорации Wonderware, обладающих функцией метки времени и качества.

Коммуникационные приложения сторонних производителей обеспечивают дополнительные возможности связи. Можно одновременно получать данные со множества серверов ввода-вывода через различные физические каналы.

IndustrialSQL Server предназначен для высокоскоростного сбора аналоговых и дискретных данных, на порядок превосходя стандартные реляционные СУБД.

## Структуры данных

Система IndustrialSQL Server снабжена двумя сконфигурированными базами данных: *Holding* и *Runtime*. База данных *Holding* используется для временного хранения данных, импортируемых в IndustrialSQL Server. База данных *Runtime* отражает функциональные особенности IndustrialSQL Server и является комбинацией стандартных и специализированных таблиц SQL Server, так называемых таблиц расширения, которые предназначены для доступа к сохраненным в реальном времени данным. Версии этих специализированных таблиц для OLE DB называются удаленными таблицами.

Таблицы SQL Server в базе данных *Runtime*:

- Логически сохраняют данные конфигурации, событий, сводок, архивов и связанные с ними производственные данные.
- Обращаются к данным, которые физически находятся на жестком диске SQL Server в файлах устройств данных.
- Обеспечивают полную поддержку языка Transact-SQL — расширения стандартного языка SQL компании Microsoft.

Таблицы расширения в базе данных *Runtime* и удаленные таблицы, предоставляемые поставщиком данных InSQL OLE DB:

- Логически сохраняют архивные данные и производственные данные в реальном времени ("живые" таблицы).
- Обращаются к данным, которые физически находятся в архивных блоках (файлах) жесткого диска.
- Полностью поддерживают запросы, содержащие расширения Industrial-SQL.
- С некоторыми ограничениями поддерживают запросы Transact-SQL.
- В целях повышения производительности запросов представляют одни и те же архивные данные в виде таблиц разных форматов.

Для облегчения доступа к данным созданы таблицы с групповой структурой, представления и хранимые процедуры. Целостность данных обеспечивается триггерами, а аналоговые и дискретные таблицы, равно как и "живые" таблицы расширения дополнительно защищены ограничением записи на сервер ввода-вывода.

Подробнее об этом см. в Справочном руководстве по IndustrialSQL Server.

## Хранение данных

IndustrialSQL Server сохраняет аналоговые, дискретные и текстовые тэги, полученные с серверов ввода-вывода, а также сложные тэги и тэги событий. Другие типы данных можно сохранять в пользовательских таблицах на сервере SQL.

Циклическое сохранение основано на промежутках времени. В параметрах сохранения можно задать все тэги вплоть до разрешения в 1000 мс, обеспечивая тем самым высокую точность представления данных, полученных с высокоскоростных устройств. Данные, изменяющиеся чаще, чем раз в 1000 мс, можно сохранять в пользовательских таблицах на сервере SQL.

Сохранение путем исключения, называемое также дельта-хранением, производится только при изменении значения тэга. Значение сохраняется вместе с датой и временем изменения. Для каждого тэга можно задать мертвые зоны времени и значения, однако для сохранения целостности данных по умолчанию они равны нулю.

Аналоговые данные можно сохранять как циклическим, так и дельта-методом. Дискретные и текстовые данные сохраняются дельта-методом.

## Области хранения

Архивные данные сводок, текстовых значений и событий сохраняются в таблицах SQL Server. Большие объемы высокоскоростных аналоговых и дискретных архивных данных сохраняются в файлах на жестком диске, так называемых архивных блоках. Текущие значения аналоговых, дискретных и строчных тэгов сохраняются в области памяти, называемой активным образом.

Подробнее об этом см. в главе 4 "Управление системой хранения".

Компактный формат хранения и алгоритм сжатия без потерь, используемые для хранения данных в архивных блоках, снижают требования к дисковому пространству до двух процентов от того, что требуется в стандартных реляционных СУБД. Архивные блоки создаются при запуске, при повторной инициализации хранения, вручную с помощью программы InSQL Control, а также через заданные интервалы времени для удобства управления данными. Сохранение архивных блоков и управление ими можно осуществлять в одной из четырех областей хранения:

1. Циклическая область является основной для хранения архивных данных. Когда область циклического хранения заполняется, самый старый архивный блок перемещается в другую область, если таковая определена. Следует регулярно делать резервные копии данных из области циклического хранения.
2. Можно задать альтернативную область хранения, чтобы избежать переполнения циклической области и застраховаться от проблем при создании резервных копий. В этом случае архивные блоки последовательно перемещаются в альтернативную область хранения, а при переходе через критический порог дискового пространства удаляются.

---

**Примечание.** В настоящее время поддерживается только одна альтернативная область хранения.

---

3. Области постоянного хранения используются для хранения критических данных (например, состояний реактора), которые нельзя перезаписывать. Как правило, области постоянного хранения представляют собой устройства хранения данных с однократной записью, такие как дисковод WORM.
4. Области буферного хранения используют для временного хранения, например, при восстановлении из резервной копии.

Можно создать комбинацию из области циклического хранения и альтернативных областей для хранения оперативных данных за период от недели до всего срока существования предприятия. IndustrialSQL Server может получать данные из всех четырех областей хранения.

Подробнее об этом см. в главе 4, "Управление системой хранения".

## Данные конфигурации

Данные конфигурации сохраняются в таблицах конфигурации базы данных *Runtime*. Таблицы конфигурации могут заполняться данными из файлов конфигурации InTouch. Можно также вручную вводить данные конфигурации при помощи программы InSQL Configure.

Конфигурация со временем меняется; добавляются и удаляются тэги, меняются их определения и технические параметры. Целостность архивных данных сохраняется, поскольку данные конфигурации предприятия доступны на всем протяжении существования предприятия.

## Производственные данные

IndustrialSQL Server гладко интегрирует данные реального времени и архивные данные с данными конфигурации, событий и сводок. Во многих приложениях производственные данные, полученные не через серверы ввода-вывода, также интегрируются с данными предприятия.

Характер производственных данных зависит от типа предприятия или процесса. К полезным данным, подлежащим включению в базу данных IndustrialSQL Server, относятся данные по смене (смена операторов), по техническому обслуживанию, по пакетам (рецептуры, пакеты и т.д.), по отслеживанию (отслеживание рабочего процесса), по стоимости (стоимость сырья), по качеству, а также данные планирования. Поддержка производственных данных осуществляется путем введения в базу данных пользовательских таблиц.

Включение этих данных в базу IndustrialSQL Server обеспечивает подлинную интеграцию управленческих и производственных данных и позволяет глубже понять взаимосвязь между производством, оплатой труда и себестоимостью.

Растет количество решений сторонних производителей, объединяющих дополнительные производственные данные с IndustrialSQL Server, на вертикальных рынках — от пивоварен до алмазных копий.

## Выборка данных

Для клиентских приложений IndustrialSQL Server представляется как сервер Microsoft SQL. Сервер базы данных IndustrialSQL Server получает запросы SQL, а затем находит, обрабатывает и возвращает данные. Если используемые клиенты поддерживают OLE DB, то запросы обрабатывает поставщик данных InSQL OLE DB.

Доступ к аналоговым архивным данным, дискретным архивным данным и "живым" данным осуществляется через таблицы расширения (удаленные). Язык Transact-SQL корпорации Microsoft расширен языком Industrial-SQL корпорации Wonderware, предназначенным для получения данных из таблиц расширения (удаленных), обеспечивая тем самым поддержку временной последовательности. В эту поддержку входят:

- Циклический и дельта-метод выборки. При циклическом методе возвращаются равномерно распределенные данные с указанным разрешением. При выборке дельта-методом данные возвращаются при каждом изменении значения тэга.
- Непрерывные запросы к таблицам, хранящим данные реального времени ("живые" таблицы). При непрерывном запросе значения данных автоматически возвращаются при каждом изменении значений в реальном времени.

---

**Примечание.** Непрерывные запросы не поддерживаются, если используется поставщик данных InSQL OLE DB или OPEN QUERY.

---

Сочетание таблиц SQL Server и таблиц расширения (удаленных) дает прекрасную возможность доступа к значимым данным, хранящимся в базе данных. Поскольку IndustrialSQL Server является реляционной базой данных, запросы могут связывать данные из разных таблиц для большей эффективности. Вот несколько примеров возможных запросов к базе данных IndustrialSQL Server:

- Среднее значение вибрации двигателя каждый день на протяжении последнего месяца при загрузке, превышающей величину  $x$ .
- Аннотация к дискретному тэгу, заархивированному на дисковом WORM шесть месяцев назад.
- Предел аналогового тэга в контексте нормального производственного режима. Предел того же аналогового тэга в контексте ускоренного производственного режима.
- Значения 50 указанных аналоговых тэгов в момент времени, когда значение  $x$  превышало 10.
- Путь к месту хранения указанного тэга.
- 20 равномерно распределенных значений данных, выбранных из всех значений, сохраненных для аналогового тэга между 8:00 и 8:30 утра 15 июля 1999 года.
- Все значения данных с 20-минутным интервалом, выбранных из всех значений, сохраненных для аналогового тэга между 8:00 и 8:30 утра 15 июля 1999 года.
- Все значения аналогового тэга, сохраненные 8 января 1999 года, где значение аналогового тэга изменилось на 10 единиц измерения. Данные для этого аналогового тэга сохранялись, если значение изменялось на 5 единиц измерения.
- Все значения тэгов, связанных с перегревом котла от 8 января 1999 года. [Подробнее об этом см. в Главе 6, "Выборка данных".](#)

## Система управления событиями

К производственным событиям относятся самые различные события — от начала и завершения работы устройств до смены персонала и партий или действий операторов. Определение и запись этих событий придают смысл непрерывным данным. В ответ на события могут предприниматься определенные действия. Система управления событиями позволяет определять события и назначать для них действия.

Например, определение перегрева котла может привести к созданию отчета по смене, определение прихода новой смены – к обновлению веб-страницы по продукции, определение производственной неполадки – к отправке сообщения по электронной почте в службу техобслуживания и так далее. IndustrialSQL Server распознает события, применяя детекторы событий по отношению к архивным данным, а затем выполняет запланированное для события действие.

Система управления событиями может также создавать сводки, основанные на произошедшем событии. Сводки создаются путем выборки тэгов и выполнения над ними вычислений.

[Подробнее об этом см. в главе 7, "Система управления событиями".](#)

## Безопасность

IndustrialSQL Server обладает всеми свойствами безопасности Microsoft SQL Server. Система безопасности SQL Server защищает сервер и хранящиеся на нем данные. В системе безопасности SQL Server всем пользователям присвоены имена и пароли (учетные записи) для доступа к базе данных. Действия, которые пользователь может выполнять с базой данных, основаны на информации его учетной записи. Эти действия, или привилегии, определяют следующее:

- Кто может получать доступ к данному серверу.
- Какие административные задачи разрешены каждому пользователю.
- К каким базам данных, объектам баз данных и элементам данных имеет доступ каждый пользователь.
- Кто может изменять информацию в базах данных.
- У кого есть право использовать определенные выражения, представления и хранимые процедуры языка Transact-SQL.

---

**Внимание!** Очень важно иметь пароль системного администратора на сервере Microsoft's SQL. При отсутствии пароля корпорация Wonderware сохраняет за собой право отказывать в технической поддержке.

---

[Подробнее об этом см. в главе 13, "Управление системой безопасности".](#)

## Резервное копирование

IndustrialSQL Server отслеживает резервирование архивных блоков и местонахождение резервных копий. Для упрощения резервирования и восстановления этих данных можно использовать утилиту резервного копирования Windows NT.

Резервные копии архивных блоков можно также делать при помощи процедуры **xp\_DiskCopy**.

Для резервного копирования баз данных используется Enterprise Manager системы Microsoft SQL Server.

Подробнее об этом см. в главе 9 "Резервное копирование данных и баз данных".

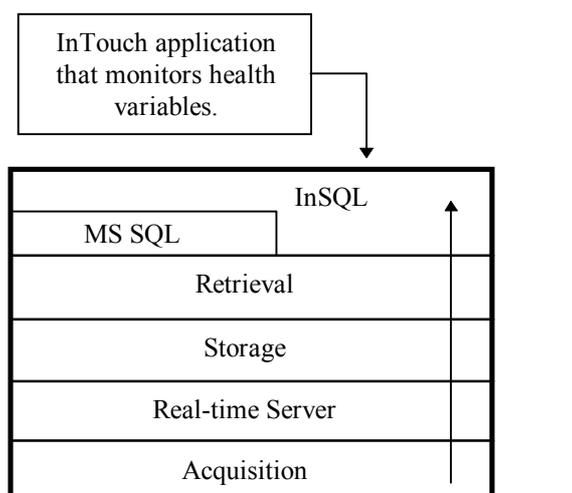
## Мониторинг и диагностика системы

Общее состояние системы IndustrialSQL Server контролирует специальный системный драйвер SYSDRV.EXE. Этот драйвер позволяет осуществлять дистанционный контроль за текущим и архивным состоянием IndustrialSQL Server и предупреждать пользователей о проблемах в системе.

Критические переменные системы (производительность, ошибки, количество подключенных пользователей, свободное пространство на диске и т.п.), а также счетчики времени (подобно тэгам предприятия) отслеживаются и сохраняются этим драйвером так же, как тэги предприятия.

Драйвер мониторинга системы в сочетании с инструментарием администратора Windows NT 4.0 (в том числе Event Viewer, Performance Monitor, Remote Access Admin и Windows NT Diagnostics) формируют ядро системы мониторинга и диагностики IndustrialSQL Server.

Мониторинг общего состояния системы производится с "верхушки" системы, проверяя работоспособность всех ее уровней, начиная с устройств сбора данных и далее вверх по всей структуре:



Например, можно написать приложение InTouch, которое будет отслеживать переменные нормального состояния системы IndustrialSQL Server.

Более подробно об этом см. в главе 8, "Мониторинг системы".

## Инструментарий администратора

Администрирование системы IndustrialSQL Server подразумевает управление запуском и выключением системы, управление конфигурацией баз данных, мониторинг работы, управление регистрацией ошибок и управление дисками. В дополнение к инструментарию корпорации Wonderware, специально разработанному для настройки IndustrialSQL Server, данная система полностью интегрирована с инструментами администратора, входящими в состав Microsoft SQL Server и Microsoft Windows NT.

Ниже перечислены инструменты Wonderware, предназначенные для выполнения задач системного администрирования:

- **InSQL Configure**  
Редактор конфигурации на основе Windows, который можно использовать для обзора объектов в системе, изменения конфигурации предприятия, настройки системы управления событиями и импортирования данных из InTouch.  
[Подробнее об этом см. в главе 2, "InSQL Configure"](#).
- **InSQL Control**  
Эта программа используется для запуска и выключения системы хранения IndustrialSQL Server и самого сервера. Из этой программы можно также контролировать статус серверов ввода-вывода.  
[Подробнее об этом см. в главе 5, "InSQL Control"](#).
- **Системный драйвер IndustrialSQL Server**  
Этот драйвер генерирует необходимые переменные состояния системы способом, сходным с получением обычных переменных предприятия.

Далее перечислены инструменты Microsoft, предназначенные для системного администрирования:

- **Microsoft SQL Server Service Manager**  
Предназначен для запуска и отключения служб Windows NT. Это необходимо для запуска или отключения компонента Microsoft SQL Server в структуре IndustrialSQL, а также для запуска или отключения самой системы IndustrialSQL Server.
- **Microsoft SQL Server Enterprise Manager**  
Используется для управления серверами SQL в системе предприятия. Набор функций позволяет администратору создавать и удалять базы данных и устройства, изменять их размеры, управлять сохранением резервных копий, извлечением данных из базы и т.д.
- **Microsoft SQL Server Client Network Utility**  
Используется для настройки информации о соединении клиент-сервер на машинах клиента.
- **Microsoft Registry Editor**  
Редактор реестра Windows.
- **Microsoft Windows Explorer**  
Предназначен для общего управления диском и контроля за регистрацией ошибок.

- **Утилита Microsoft Backup**  
Предназначена для сохранения резервных копий или восстановления архивных блоков с использованием цифровых магнитных носителей.
- **Microsoft Event Viewer**  
Система IndustrialSQL Server регистрирует ошибки в журнале событий Windows NT. Программа Event Viewer позволяет просмотреть этот журнал.
- **Microsoft Performance Monitor**  
Приложение, позволяющее отслеживать использование ресурсов отдельными компонентами и процессами программ. Полезно для выявления и устранения причин замедления работы системы.

## Инструментарий клиента

Одно из главных преимуществ IndustrialSQL Server – широкий диапазон клиентских приложений, которые могут работать с этой системой. Пользователи могут получать доступ к данным через выбранное ими программное обеспечение, а разработчики могут создавать по их заказу приложения с требуемыми функциями.

Для обеспечения доступа к информации IndustrialSQL Server можно использовать клиентские приложения двух категорий:

- Клиентские приложения Wonderware
- Любые инструменты запросов сторонних разработчиков, умеющие обращаться к данным через SQL/ODBC

Подробнее об использовании клиентских инструментов с сервером IndustrialSQL см. в *Руководстве по клиентским инструментам IndustrialSQL Server*.

## Клиентские инструменты Wonderware

Корпорация Wonderware поставляет ряд клиентских инструментов, отвечающих различным требованиям к представлению и анализу данных. Существуют клиентские приложения IndustrialSQL Server для определения набора данных, анализа трендов и просмотра данных в реальном времени. Эти инструменты избавляют пользователя от необходимости быть знакомым с SQL и обеспечивают интуитивно понятные интерфейсы типа «point-and-click» для доступа, анализа и построения графиков как для текущих, так и для архивных данных временной последовательности. В настоящее время в пакет FactoryOffice входят следующие приложения для IndustrialSQL Server:

- InSQL Trend
- InSQL QuickLook
- InSQL Report
- InSQL VectorMaster и VectorViewer
- Industrial Workbook

---

Помимо этого, есть возможность поместить перечисленные ниже управляющие элементы ActiveX в любой контейнер ActiveX и использовать их для получения информации из базы данных IndustrialSQL Server.

- ActiveDataGrid
- ActiveGraph
- ActiveTagBrowser
- ActiveTimeSelector
- ActiveTrend

---

**Примечание.** IndustrialSQL Server Server можно также использовать с другими клиентскими приложениями Wonderware, такими как InBatch, InTrack и InTouch.

---

Более подробно о клиентских инструментах Wonderware см. в *Руководстве по клиентским инструментам IndustrialSQL Server*.

## Инструменты сторонних разработчиков

- Существуют многочисленные коммерческие инструменты запросов и отчетов, обеспечивающие высококачественные, удобные интерфейсы для работы с данными с использованием SQL. Любые клиентские инструменты с интерфейсом для Microsoft SQL или ODBC пригодны для доступа к данным IndustrialSQL Server и генерации отчетов.

# InSQL Configure

В настоящей главе описано использование программы InSQL Configure для просмотра и редактирования информации об объектах баз данных. К объектам баз данных относятся тэги, серверы ввода-вывода, темы, пользователи и т.д.

Если редактирование производится при помощи программы InSQL Configure, по его завершении может потребоваться перезагрузка системы.

## Содержание

- О программе InSQL Configure
- Запуск программы InSQL Configure
- Как ориентироваться в главном окне
- Настройка информации о сервере
- Настройка информации тэгов
- Настройка информации о местах хранения
- Настройка серверов ввода-вывода
- Повторная инициализация
- Отслеживание изменений
- Управление дисковым пространством

## О программе InSQL Configure

Программа InSQL Configure является клиентским приложением для управления конфигурацией системы IndustrialSQL Server. Программа InSQL Configure позволяет:

- Импортировать тэги в систему InSQL из InTouch.
- Просматривать и редактировать детальные свойства тэгов, импортированных из InTouch.
- Добавлять, редактировать и удалять объекты IndustrialSQL, такие как серверы ввода-вывода, темы и тэги.
- Добавлять, редактировать и удалять единицы измерения и дискретные пары сообщений.
- Добавлять, редактировать и удалять определения событий.
- Настраивать необходимые агрегации в соответствии с определяемой пользователем схемой.
- Настраивать пути к месту хранения данных.
- Просматривать информацию о тэге состояния системы.
- Создавать группы в общедоступных и личных папках.
- Перезагружать систему.
- Запускать процесс отслеживания изменений.

Темы и данные конфигурации можно импортировать в систему IndustrialSQL Server из InTouch. Данные конфигурации и тем импортируются при помощи функции Tag Importer, входящей в состав InSQL. При запущенной службе Tag Importer можно выбрать из текущего приложения InTouch категорию тем (все, ввод-вывод предприятия, памяти и т.д.), которую нужно импортировать на сервер IndustrialSQL. Вся информация об определении тэгов, которая содержалась в файле TAGNAME.X приложения InTouch, будет теперь сохранена на сервере IndustrialSQL, равно как и другие данные конфигурации.

[Подробнее об импорте тэгов см. в главе 3, "Интеграция InTouch" данного руководства администратора.](#)

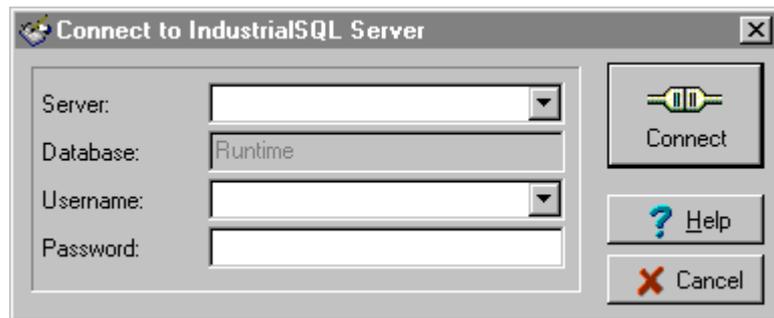
## Запуск программы InSQL Configure

Для того, чтобы производить изменения в системе IndustrialSQL Server, у вас должны быть привилегии для работы с базами данных. Вы должны быть либо членом группы "wwAdministrators" либо владельцем базы данных. Если вы не регистрируетесь с привилегиями администратора, то ваши полномочия будут ограничены. Например, у вас не будет доступа к главной панели инструментов и к меню **Tools**. Если вы регистрируетесь с привилегиями пользователя ("wwUser"), вы сможете редактировать информацию при помощи программы InSQL Configure, но не сможете сохранить изменения.

➤ **Как запустить программу и зарегистрироваться на сервере IndustrialSQL:**

1. В меню **Start** выберите группу программ IndustrialSQL Server, а затем – программу InSQL Configure.

Запустится программа InSQL Configure, и автоматически появится диалоговое окно **Connect to IndustrialSQL Server**.



2. Для соединения с сервером все клиентские приложения IndustrialSQL Server требуют ввода данных учетной записи. По умолчанию автоматически появляется информация, введенная при последней успешной регистрации. При необходимости введите соответствующую информацию учетной записи.

---

**Примечание.** Учетная запись пользователя состоит из имени пользователя и пароля. Учетная запись пользователя должна включать в себя право на доступ, иначе регистрация не состоится. Подробнее об учетной записи пользователя вам может рассказать системный администратор.

---

### Server

Имя сервера IndustrialSQL, с которым вы хотите соединиться. По умолчанию в этом поле появляется список всех серверов IndustrialSQL, с которыми за последнее время были произведены успешные соединения.

**Database**

Данное поле, предназначенное только для чтения, отображает имя базы данных на сервере, к которой вы хотите соединиться. По умолчанию появляется база данных *Runtime*.

**Username**

Введите присвоенное вам имя пользователя IndustrialSQL Server. Если системный администратор не присвоил вам имя и пароль, можно использовать учетную запись пользователя «по умолчанию», которая автоматически создается при установке IndustrialSQL Server. По умолчанию в этом поле появляется список тех имен пользователей, при помощи которых за последнее время были произведены успешные соединения.

Информацию об учетной записи пользователя «по умолчанию» см. в главе 1, "Введение", справочного руководства по серверу *IndustrialSQL*.

**Password**

Ведите пароль, соответствующий данному имени пользователя. В отдельных случаях это поле может быть пустым.

3. Нажмите **Connect**, чтобы зарегистрироваться, либо **Cancel**, чтобы выйти.

Если указанный сервер IndustrialSQL обнаружен в сети, и введенные имя пользователя и пароль действительны, диалоговое окно **Connect to IndustrialSQL Server** закроется, и появится главное окно программы.

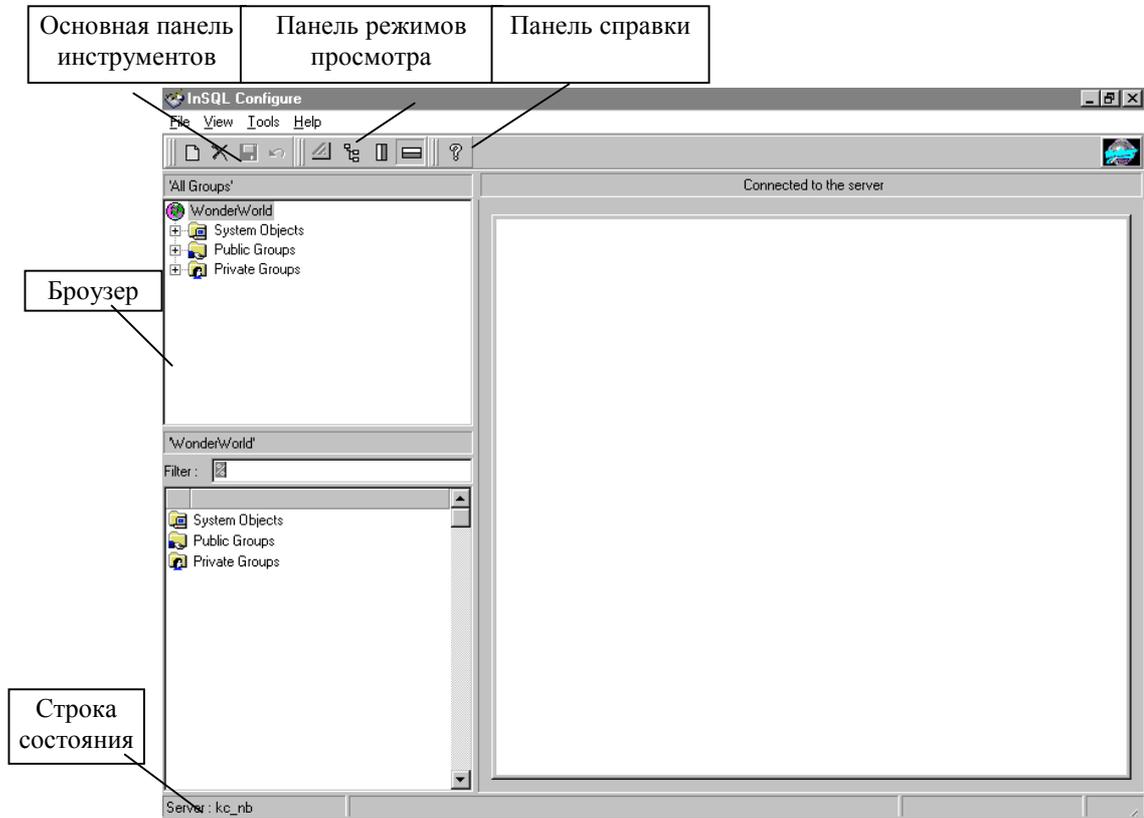
Если вы получили сообщение об ошибке "Не могу найти сервер SQL либо соединиться с ним", это означает, что введено неправильное имя сервера либо нарушено сетевое соединение между рабочей станцией и сервером.

Если получено сообщение об ошибке "Неопознанное имя пользователя или пароль. Регистрация отменяется", это означает, что имя пользователя или пароль не подходят для указанного IndustrialSQL Server.

Кроме того, убедитесь, что имя пользователя и пароль напечатаны в правильном регистре, в том случае, если база данных учитывает регистр.

## Как ориентироваться в главном окне

При запуске программы InSQL Configure появляется главное окно:



## Панели инструментов InSQL

В них входят основная панель инструментов, панель режимов просмотра, панель справки и логотип корпорации Wonderware.

### ➤ Как скрыть панели инструментов:

- В меню **View** выберите **Toolbars**, чтобы за командой следовало слово **[Hidden]**.

## Основная панель инструментов

Используйте кнопки этой панели, чтобы сохранить либо отменить изменения в конфигурации объектов.



Ниже приводятся описания кнопок данной панели:

Кнопка	Описание
	Создать новый объект (аналоговый, дискретный, текстовый, событие, сервер ввода-вывода, тема, единицу измерения либо пару сообщений).
	Удалить объект.
	Сохранить изменения.
	Отменить изменения, произведенные со времени последнего сохранения.

## Панель инструментов режимов просмотра

Кнопки данной панели предназначены для выполнения команд, относящихся к отображению элементов программы.



Кнопки панели режимов просмотра описаны ниже:

Кнопка	Описание
	Отобразить/скрыть строку состояния.
	Отобразить/скрыть броузер.
	Отобразить броузер вертикально.
	Отобразить броузер горизонтально.

## Панель справки и логотип

Для запуска файла справки используйте кнопку «справка» данной панели.



По умолчанию в верхнем правом углу отображается логотип корпорации Wonderware.



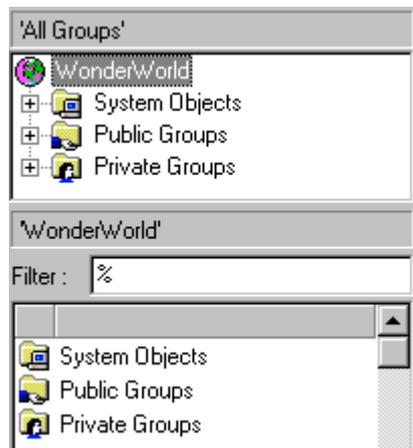
## Строка состояния

Строка состояния дает возможность просматривать статус соединения с сервером IndustrialSQL, а также любые другие сообщения о статусе, отправленные клиентом:



## Броузер

Броузер представляет собой двухпанельное графическое представление групп объектов в базе данных. Броузер отображает все объекты системы, например, тэги, которые видны данному отдельному клиенту. Каждый объект базы данных является пунктом списка на броузере. Броузер отображается по умолчанию. Чтобы скрыть броузер, в меню **View** выберите **Browser** и нажмите **Hidden**, чтобы снять флажок. Либо нажмите кнопку  на панели инструментов.



Можно перемещаться по броузеру так же, как и по любому стандартному приложению-броузеру на основе Windows NT. Правая кнопка мыши используется для расширения уровня, а левая – для действий на выбранном уровне. При нажатии на клавиатуре кнопки Home будет выбран верхний пункт, при нажатии кнопки End — нижний.

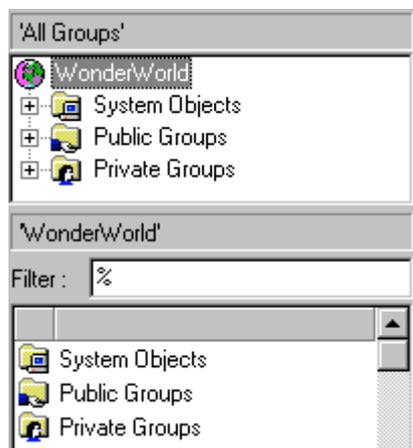
Броузер отображает объекты для следующих категорий:

Категория	Описание
System Objects	Все объекты базовой конфигурации серверной системы IndustrialSQL, такие как тэги, серверы ввода-вывода, заданные единицы измерения, места хранения данных и т.д..
Public Groups	Все объекты, видимые всеми клиентами. Если вы располагаете привилегиями администратора, можно создавать, переименовывать и удалять группы в папках общих групп. Все новые подпапки, создаваемые в папках общих групп (то есть те, которые не создаются по умолчанию), будут зеленого цвета.
Частные группы	Все объекты, видимые подключенному в настоящее время пользователю. Пользователи могут создавать, переименовывать и удалять группы в папках частных групп. Все новые подпапки, создаваемые в папках частных групп, будут синего цвета.

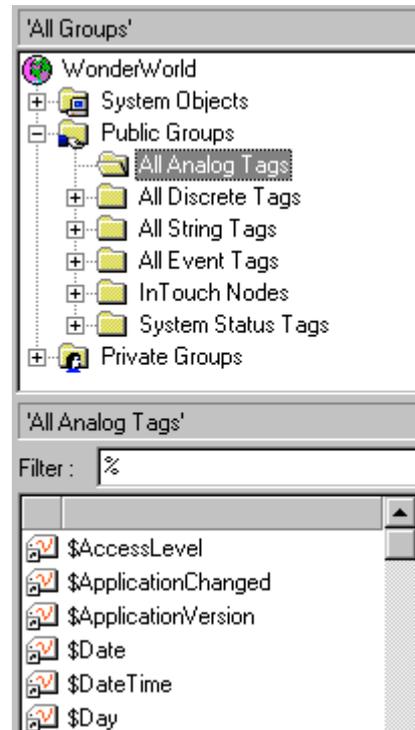
Подробнее об этом см. "Редактирование групп" ниже в этой главе.

## Вид броузера

Броузер можно представить двояко: вертикально или горизонтально. По умолчанию программа конфигурации InSQL запускается с горизонтальным положением броузера:



На верхней панели отображается общий список броузера (группы). При выборе группы или объекта в верхней панели броузера на нижней панели появятся все объекты, стоящие ниже в групповой иерархии.



Для горизонтального отображения броузера в меню **View** выберите **Browser** и нажмите **Horizontal**.

Для выбора способа отображения броузера можно использовать также кнопки  и  на панели инструментов.

## Использование фильтра броузера

Для поиска тэга на панели объектов броузера введите имя тэга в поле **Filter**. По мере ввода в броузере начнут появляться тэги, начинающиеся с первой введенной буквы. Можно использовать символ **%** для подстановки ряда символов. Например, чтобы найти все тэги, заканчивающиеся на "level", введите "%level".

## Редактирование групп

Контекстное меню браузера содержит команды для добавления групп в иерархию и добавления к ним объектов. (Доступ к этому меню осуществляется нажатием правой кнопкой мыши в окне браузера). Новая группа добавляется точно так же, как и новая папка в Проводнике Windows. Например, можно создать группу "BoilerTags" внутри существующей группы "Private Groups". Можно также удалять, вырезать, копировать, вставлять и перетаскивать объекты из одной папки в другую.

### Добавление группы

➤ **Как добавить группу:**

1. Нажмите правой кнопкой мыши на папке, внутри которой хотите создать группу.
2. Появится контекстное меню.
3. Нажмите **Add Group**.
4. В браузере появится новая папка.
5. Введите имя папки и нажмите **Enter**.

### Добавление тэга в группу

Если вы добавляете тэги в новую группу, оригинальная ссылка будет по-прежнему появляться в системной группе по умолчанию. Любой тэг может принадлежать любому количеству групп, и любая группа может содержать любое количество тэгов.

➤ **Как добавить тэг в группу:**

1. Выберите группу, в которую хотите добавить тэг.
2. Выполните одно из следующих действий:
  - Перетащите нужный тэг с панели объектов в папку.
  - Нажмите правой кнопкой мыши на тэге на панели объектов. При помощи команд **Copy** и **Paste** контекстного меню скопируйте тэг в выбранную папку.

## Удаление ссылки на группу или тэг

Если вы хотите удалить тэг из системы, нужно его выбрать в группе Topics данного сервера. При удалении тэга все ссылки на него как в общедоступных, так и в частных папках также удаляются.

Когда вы удаляете частную группу, папка этой группы и все ссылки на тэги будут также удалены. Сами же тэги удалены не будут, оригинальные ссылки по-прежнему будут появляться в системной группе по умолчанию. Нельзя удалить общедоступные папки (желтые) или содержащиеся в них ссылки на тэги.

### ➤ Как удалить группу или объект:

1. Выберите в браузере группу или объект.
2. Удалите выбранное одним из следующих способов:
  - Нажмите правой кнопкой мыши на группе или объекте. В появившемся контекстном меню нажмите **Delete**.
  - На панели инструментов нажмите кнопку .

## Переименование группы или ссылки на тэг

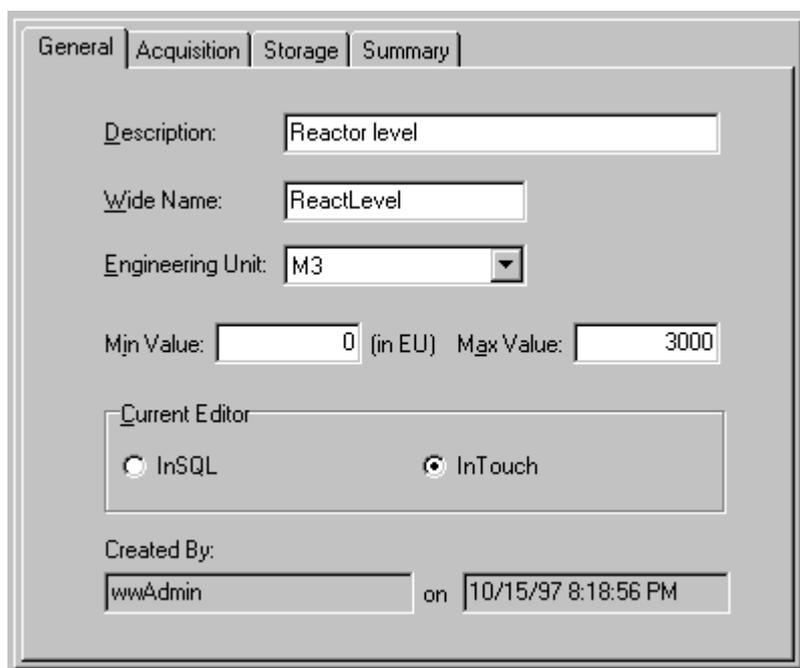
Можно переименовать группу или объект, созданные вами в браузере. Однако нельзя переименовать общедоступные папки (желтые) или ссылки на тэги.

### ➤ Как переименовать группу или объект:

1. Нажмите правой кнопкой мыши на группе или объекте в браузере.
2. В появившемся контекстном меню нажмите **Rename**.
3. Введите новое имя папки и нажмите **Enter**.

## Панель свойств

Программа InSQL Configure содержит панель свойств, или контекстную панель, в которой можно легко настроить определенные характеристики объекта, выбранного в браузере. Например, если выбран аналоговый тэг, появится панель свойств аналоговых тэгов.



The screenshot shows a dialog box with four tabs: General, Acquisition, Storage, and Summary. The 'General' tab is active. It contains the following fields and controls:

- Description:** A text box containing 'Reactor level'.
- Wide Name:** A text box containing 'ReactLevel'.
- Engineering Unit:** A dropdown menu showing 'M3'.
- Min Value:** A text box containing '0' with '(in EU)' next to it.
- Max Value:** A text box containing '3000'.
- Current Editor:** A section with two radio buttons: 'InSQL' (unselected) and 'InTouch' (selected).
- Created By:** A text box containing 'wwAdmin'.
- Created On:** A text box containing '10/15/97 8:18:56 PM'.

Просмотрите экраны этой панели, нажимая на соответствующие вкладки.

## Как показать или скрыть программные элементы

Программные элементы, которые можно скрыть — это панели инструментов, браузер и строка состояния.

- **Как показать программный элемент:**
  1. В меню **View** выберите соответствующую команду (например, **Toolbar** для панели инструментов) так, чтобы рядом с ней появилась галочка и слово **[Shown]**.
  2. Программный элемент появится в окне.
- **Как скрыть программный элемент:**
  1. В меню **View** выберите соответствующую команду (например, **Toolbar** для панели инструментов) так, чтобы рядом с ней исчезла галочка и появилось слово **[Hidden]**.
  2. Программный элемент исчезнет из окна.

## Просмотр информации о программе

Можно просмотреть информацию о программе InSQL Configure, такую как номер версии, авторское право и лицензию.

➤ **Как просмотреть информацию о программе:**

1. В меню **Help** выберите **About**.

Появится диалоговое окно **About Factory Office**.



2. В диалоговом окне показаны сведения о версии и авторском праве программе.
3. Для просмотра лицензии нажмите **View License**.  
Дополнительную информацию о лицензии см. в *Руководстве администратора системы FactorySuite*.
4. Нажмите **OK**.

## Настройка информации о сервере

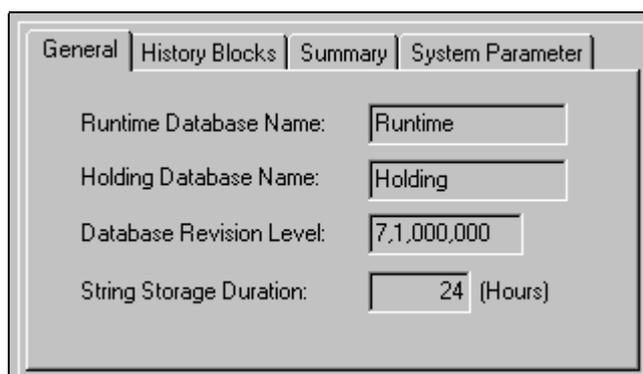
InSQL Configure позволяет просматривать и редактировать глобальную информацию о системе. Можно просмотреть и отредактировать следующие сведения для IndustrialSQL Server:

- Общие свойства
- Архивные блоки
- Параметры сводок
- Именные системные параметры

### Просмотр общих свойств сервера

➤ **Как просмотреть общие свойства сервера:**

1. В окне браузера разверните группу **System Object**, затем разверните группу **Servers** и выберите сервер, общие свойства которого требуется просмотреть.
2. Выберите вкладку **General**.



3. Появятся свойства сервера в режиме только для чтения.

**Runtime Database Name**

Имя базы данных *выполнения*. В этой базе хранятся все объекты IndustrialSQL (таблицы, хранимые процедуры, представления и т.д.).

**Holding Database Name**

Имя базы данных *накопления*. Эта база используется для временного хранения данных словаря тэгов InTouch перед импортированием тэгов в IndustrialSQL Server.

**Database Revision Level**

Номер версии базы данных *выполнения*.

**String Storage Duration**

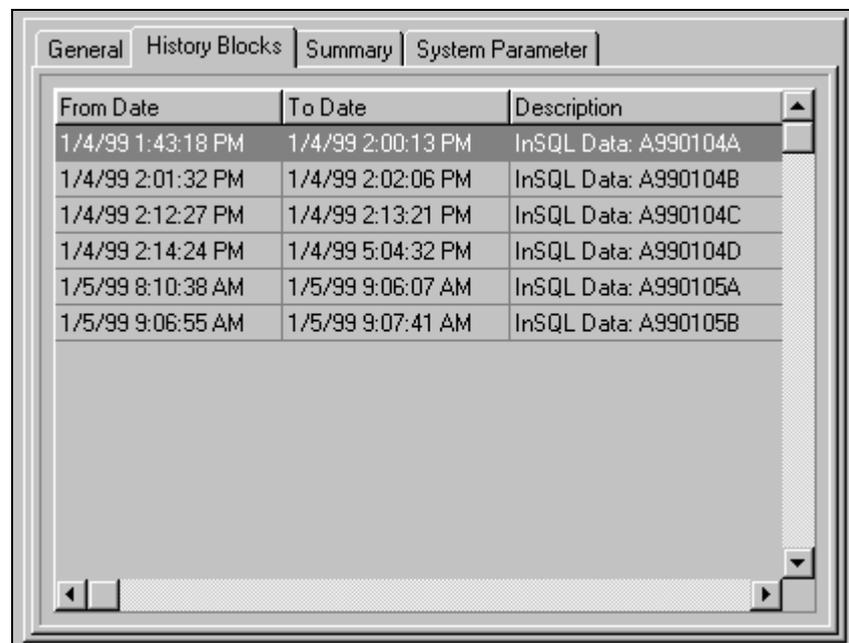
Период времени, в течение которого IndustrialSQL Server будет хранить текстовые значения.

## Просмотр архивных блоков

Архивный блок представляет собой группу файлов хранения данных, которые записываются в отдельный каталог, идентифицируемый меткой даты и буквенным суффиксом. Система IndustrialSQL Server хранит полученные аналоговые и дискретные архивные данные в блоках. Архивные блоки создаются при запуске, через заданные интервалы времени согласно настройкам в реестре или вручную с помощью программы InSQL Control или расширенной хранимой процедуры `xp_NewHistoryBlock`.

➤ **Как посмотреть архивные блоки:**

1. В браузере выберите сервер, архивные блоки которого требуется просмотреть.
2. Выберите вкладку **History Blocks**.



From Date	To Date	Description
1/4/99 1:43:18 PM	1/4/99 2:00:13 PM	InSQL Data: A990104A
1/4/99 2:01:32 PM	1/4/99 2:02:06 PM	InSQL Data: A990104B
1/4/99 2:12:27 PM	1/4/99 2:13:21 PM	InSQL Data: A990104C
1/4/99 2:14:24 PM	1/4/99 5:04:32 PM	InSQL Data: A990104D
1/5/99 8:10:38 AM	1/5/99 9:06:07 AM	InSQL Data: A990105A
1/5/99 9:06:55 AM	1/5/99 9:07:41 AM	InSQL Data: A990105B

3. В окне появится информация о каждом архивном блоке.

---

**Примечание.** Если вы соединились с сервером Microsoft SQL Server вместо IndustrialSQL Server, архивные блоки не появятся в этом окне.

---

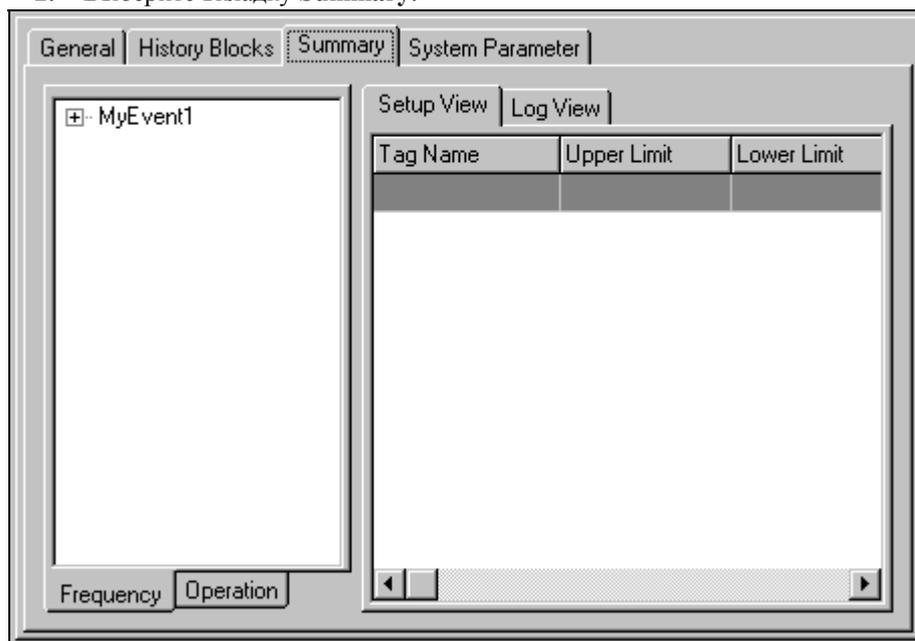
Подробнее об архивных блоках см. в главе 4, "Управление системой хранения".

## Просмотр информации о сводках

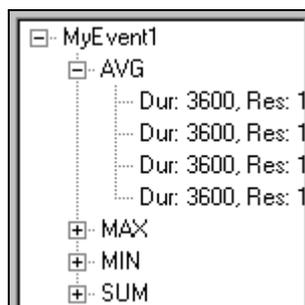
Если вы сконфигурировали операции сводок в системе управления событиями, вы сможете просмотреть информацию о них в панели свойств **Summary**. Можно также просмотреть архив сводок.

### ➤ Как просмотреть информацию о сводках:

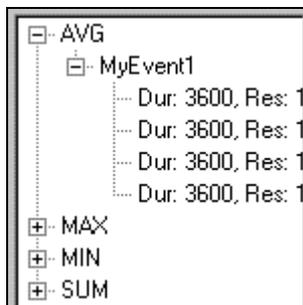
1. В браузере выберите сервер, сводки которого требуется просмотреть .
2. Выберите вкладку **Summary**.



3. Для просмотра сводок, сгруппированных по частоте (заданной для тэгов событий) откройте вкладку **Frequency**.
4. Развернув иерархическое дерево, можно просмотреть все операции сводок для определенных тэгов событий. Например:



5. Для просмотра сводок, сгруппированных по операциям (AVG, MIN, MAX, SUM) откройте вкладку **Operation**.
6. Развернув иерархическое дерево, можно просмотреть все операции сводок для определенной агрегации. Например:



7. Выберите операцию (самый нижний уровень иерархии) на вкладке **Frequency** или **Operation**.
8. Детали сводок появятся на вкладках **Setup View** и **Log View**.
9. Выберите вкладку **Setup View**, чтобы просмотреть, какие тэги сконфигурированы для выбранной операции сводки.

Tag Name	Upper Limit	Lower Limit
ReactLevel	-9999999	9999999
ReactTemp	-9999999	9999999

10. На вкладке **Setup View** имеются следующие столбцы:

**Tag Name**

Имя тэга, по которому делается сводка.

**Upper Limit**

Верхний предел допустимых значений тэга. Значения, превышающие этот предел, не берутся в расчет. По умолчанию это значение устанавливается на максимальное физическое значение тэга (MaxVal).

**Lower Limit**

Нижний предел допустимых значений тэга. Значения ниже этого предела не берутся в расчет. По умолчанию это значение устанавливается на минимальное физическое значение тэга (MinVal).

8. Нажмите **Log View**, чтобы просмотреть результаты выбранной операции сводки (если она уже выполнялась).



9. В журнале всегда присутствует запись, даже если операция не выдавала результатов (например, если для операции не были определены тэги). На вкладке **Log View** имеются следующие столбцы:

**Date**

Метка даты/времени данных сводки.

**Start Time**

Время начала сводного расчета.

**End Time**

Время окончания сводного расчета.

---

**Совет.** Если операция выполнялась слишком долго, попробуйте уменьшить разрешение или количество тэгов, входящих в сводку.

---

## Редактирование именных системных параметров

Именной системный параметр — это параметр, управляющий одним из аспектов общего поведения системы. В IndustrialSQL Server определены следующие именные системные параметры:

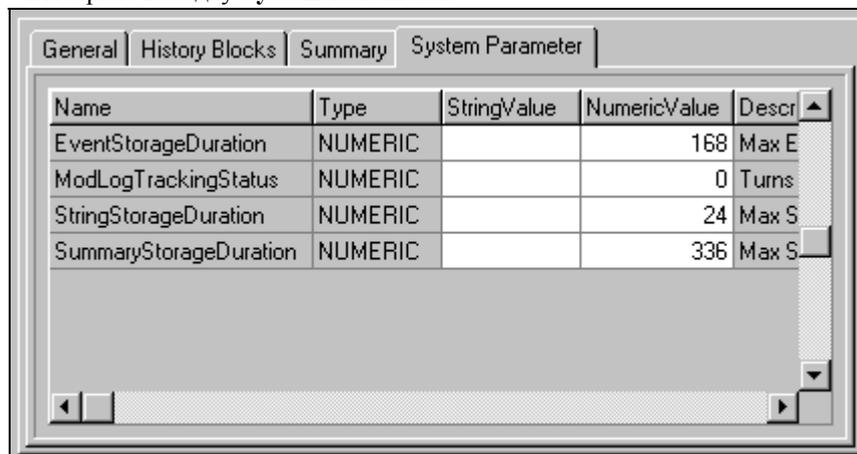
<b>Именной системный параметр</b>	<b>Описание</b>
DatabaseVersion*	Текущая версия базы данных выполнения.
EventStorageDuration	Максимальная длительность хранения значений событий.
InSQLListenOn*	Строка соединения с IndustrialSQL Server.
ModLogTrackingStatus	Определяет, включено ли отслеживание изменений.
SQLListenOn*	Строка соединения с Microsoft SQL Server.
StringStorageDuration	Максимальная длительность хранения значений текстовых тэгов.
SummaryStorageDuration	Максимальная длительность хранения значений автоматизированных сводок.
SystemVersion*	Текущая версия системы IndustrialSQL Server.

\*Не показывается в InSQL Configure.

Вы можете создать собственные именные системные параметры для IndustrialSQL Server, добавив строки к таблице *NamedSystemParameter* с помощью сценария SQL.

➤ **Как просмотреть системные параметры:**

1. В окне браузера разверните группу **System Object**, затем разверните группу **Servers** и выберите сервер, именные параметры которого требуется просмотреть.
2. Выберите вкладку **System Parameter**.



The screenshot shows a window titled "System Parameter" with four tabs: "General", "History Blocks", "Summary", and "System Parameter". The "System Parameter" tab is active, displaying a table with the following data:

Name	Type	StringValue	NumericValue	Descr
EventStorageDuration	NUMERIC		168	Max E
ModLogTrackingStatus	NUMERIC		0	Turns
StringStorageDuration	NUMERIC		24	Max S
SummaryStorageDuration	NUMERIC		336	Max S

3. В окне появятся все определенные системные параметры.
4. Вы можете изменить значения в редактируемых столбцах панели свойств.
5. Чтобы сохранить изменения, нажмите кнопку . Или в меню **File** выберите **Save**.
6. Чтобы отменить изменения, нажмите кнопку . Или в меню **File** выберите **Undo**.

## Настройка информации тэгов

Тэг — это переменная в системе IndustrialSQL Server, в которую записываются и из которой считываются значения. Каждый тэг имеет уникальное имя. В имени тэга нельзя использовать пробелы, кавычки и другие специальные символы, хотя программа InSQL Configure может позволить их ввести. Можно создать тэги следующих типов:

- Аналоговые
- Дискретные
- Текстовые
- Тэги событий

В системе IndustrialSQL Server хранится информация конфигурации каждого типа тэгов, равно как и архив значений тэгов за определенный период времени. Для тэгов событий хранятся не значения, а определения событий и ответные действия.

Подробнее о тэгах см. в *Справочном руководстве IndustrialSQL Server*.

С помощью InSQL Configure можно просматривать и редактировать информацию тэгов, импортированных в IndustrialSQL Server, настраивать дополнительные свойства, такие как сводки тэгов, или создавать новые тэги.

## Редактирование конфигурации сбора данных для тэга

Вкладка **Acquisition** содержит в основном одинаковую информацию для аналоговых, дискретных и текстовых тэгов. Однако на вкладках текстовых и дискретных тэгов отсутствуют группы **Raw Type**, **Scaling** и поля **Min Raw** и **Max Raw**.

### ➤ Как отредактировать конфигурацию сбора данных для тэга:

1. Выберите тэг в браузере.
2. Откройте вкладку свойств **Acquisition**.

The screenshot shows a configuration window with four tabs: General, Acquisition, Storage, and Summary. The 'Storage' tab is active. Under the 'Addressing' section, there are three dropdown menus: 'IO Server' with the value '\\kc\_computer1\WIEW', 'Topic Name' with 'TagName', and 'Item Name' with 'SetPoint'. Below this is the 'Raw Type' section, which contains two radio buttons: 'Integer' (which is selected) and 'Real'. To the right of these is an 'Integer Size' dropdown menu set to '32 Signed'. The 'Scaling' section contains two radio buttons: 'None' and 'Linear' (which is selected). At the bottom of the window, there are two input fields: 'Min Raw' with the value '0' and 'Max Raw' with the value '100'.

3. В списке **IO Server** выберите имя сервера ввода-вывода. В этом списке перечислены все серверы ввода-вывода, определенные в системе.
4. Конфигурация сервера ввода-вывода всегда содержит имя машины для тем DDE или SuiteLink. В среде выполнения имя машины может игнорироваться, если существует соединение DDE или SuiteLink. [Подробнее о серверах ввода-вывода см. в главе 10.](#)
5. В списке **Topic Name** выберите имя темы для тэга.  
Темы, показанные в списке, зависят от выбора сервера ввода-вывода.
6. В поле **Item Name** ведите адресную строку тэга.
7. Если редактируется текстовый или дискретный тэг, нажмите  для сохранения изменений. Или в меню **File** выберите **Save**. Чтобы отменить изменения, нажмите кнопку . Или в меню **File** выберите **Undo**.  
В противном случае, перейдите к шагу 7.
8. В группе **Raw Type** выберите числовой тип для получаемого необработанного значения.

#### **Integer**

Целое значение. При выборе этого параметра появится ниспадающий список, из которого можно выбрать размер целого числа в битах.

#### **Real**

Число с плавающей точкой (десятичное). Значение с плавающей точкой может быть в диапазоне от  $-3.4e^{38}$  до  $+3.4e^{38}$ . Все расчеты с плавающей точкой выполняются с 64-битовым разрешением, но результаты хранятся с 32-битовым.

9. В группе **Scaling** выберите тип алгоритма масштабирования из необработанных значений в единицы измерения.

При линейном масштабировании результат вычисляется методом линейной интерполяции между конечными точками.  
Подробнее см. в *Руководстве пользователя InTouch*.

10. В поле **Min Raw** введите минимальное значение для полученных необработанных значений. (Это поле не появится, если выбрать **None** в качестве типа **Scaling**.)
11. В поле **Max Raw** введите максимальное значение для полученных необработанных значений. (Это поле не появится, если выбрать **None** в качестве типа **Scaling**.)
12. Чтобы сохранить изменения, нажмите кнопку . Или в меню **File** выберите **Save**.
13. Чтобы отменить изменения, нажмите кнопку . Или в меню **File** выберите **Undo**.

## Настройка аналоговых тэгов

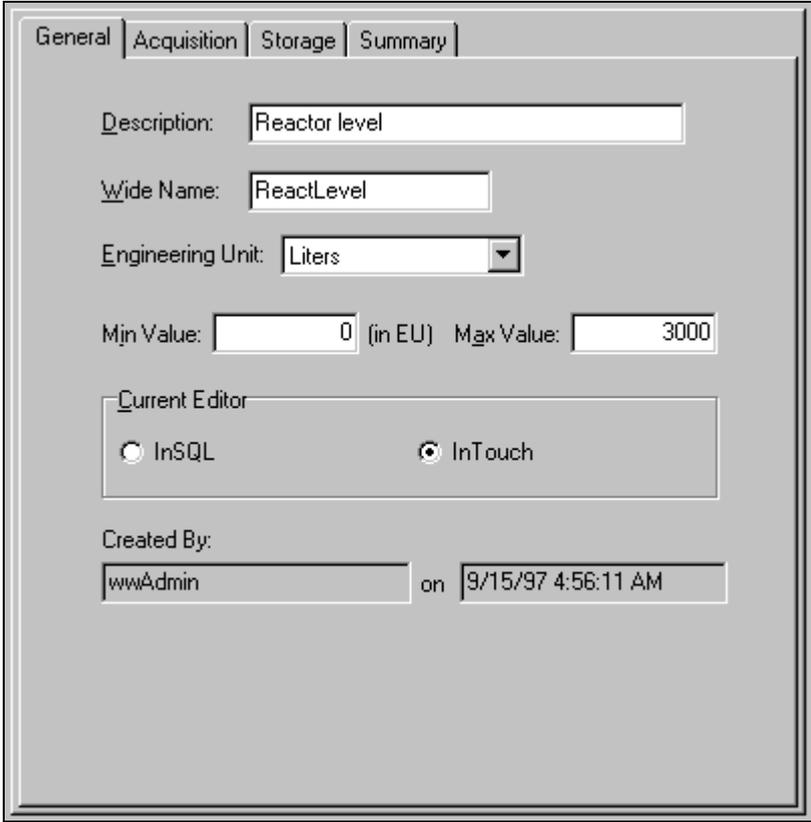
Для просмотра или редактирования конфигурации аналогового тэга нужно просто выбрать его в браузере. Информация тэга появится в соответствующей панели свойств. Можно настроить общие свойства, свойства сбора данных, свойства хранения данных и свойства сводки, а также можно добавить новый аналоговый тэг.

Информацию о настройке свойств сбора данных см. в разделе "Редактирование конфигурации сбора данных тэга" выше в этой главе.

### Редактирование общих свойств аналогового тэга

➤ **Как отредактировать общие свойства аналогового тэга:**

1. Выберите в браузере аналоговый тэг.
2. Откройте вкладку свойств **General**.



The screenshot shows a configuration window with four tabs: 'General', 'Acquisition', 'Storage', and 'Summary'. The 'General' tab is active. It contains the following fields and controls:

- Description:** A text input field containing 'Reactor level'.
- Wide Name:** A text input field containing 'ReactLevel'.
- Engineering Unit:** A dropdown menu with 'Liters' selected.
- Min Value:** A text input field containing '0' (in EU).
- Max Value:** A text input field containing '3000'.
- Current Editor:** A group box containing two radio buttons: 'InSQL' (unselected) and 'InTouch' (selected).
- Created By:** A text input field containing 'wwAdmin'.
- Created On:** A text input field containing '9/15/97 4:56:11 AM'.

3. В поле **Description** введите описание тэга.

4. В поле **Wide Name** введите версию тэга в виде допустимого для SQL Server имени столбца.

В широких таблицах (*WideTableDictionary*, *AnalogWideHistory* и *DiscreteWideHistory*) один или несколько имен тэгов используются как имена столбцов. Если из InTouch импортировано имя тэга, содержащее недопустимые символы, его следует переименовать согласно правилам для идентификаторов Microsoft SQL Server. См. документацию по Microsoft SQL Server.

5. В списке **Engineering Unit** выберите единицу измерения для аналогового тэга. Например, секунды, CPS (символов в секунду), chars (символы). [Подробнее о добавлении единиц измерения в систему см. в разделе "Добавление единицы измерения" далее в этой главе.](#)
6. В полях **MinValue** и **MaxValue** введите минимальное и максимальное значения аналогового тэга, выраженное в единицах измерения.
7. В группе **Current Editor** укажите, где будет редактироваться тэг — в IndustrialSQL Server или в InTouch.

Параметр Current Editor определяет, в какой среде будет редактироваться тэг, либо в InSQL Configure или в InTouch (через функцию импорта тэгов). Тэги, импортированные из InTouch, изначально настроены на InTouch как среду редактирования. Если с тэгом производятся изменения в InSQL Configure, то редактор тэга меняется на InSQL Configure. При повторном импортировании все изменения, сделанные с помощью InSQL Configure, сохраняются. Можно вручную сохранять InTouch в качестве редактора для повторного импортирования; однако любые изменения, сделанные в тэге с помощью InSQL Configure, будут потеряны во время повторного импортирования.

8. Чтобы сохранить изменения, нажмите кнопку . Или в меню **File** выберите **Save**.
9. Чтобы отменить изменения, нажмите кнопку . Или в меню **File** выберите **Undo**.

Следующие параметры вкладки **General** нельзя редактировать для аналогового тэга.

**Created By**

Имя пользователя или приложения, создавшего тэг. Генерируется автоматически при наполнении базы данных.

**On**

Дата создания тэга. Генерируется автоматически при наполнении базы данных.

## Редактирование свойств хранения данных аналогового тэга

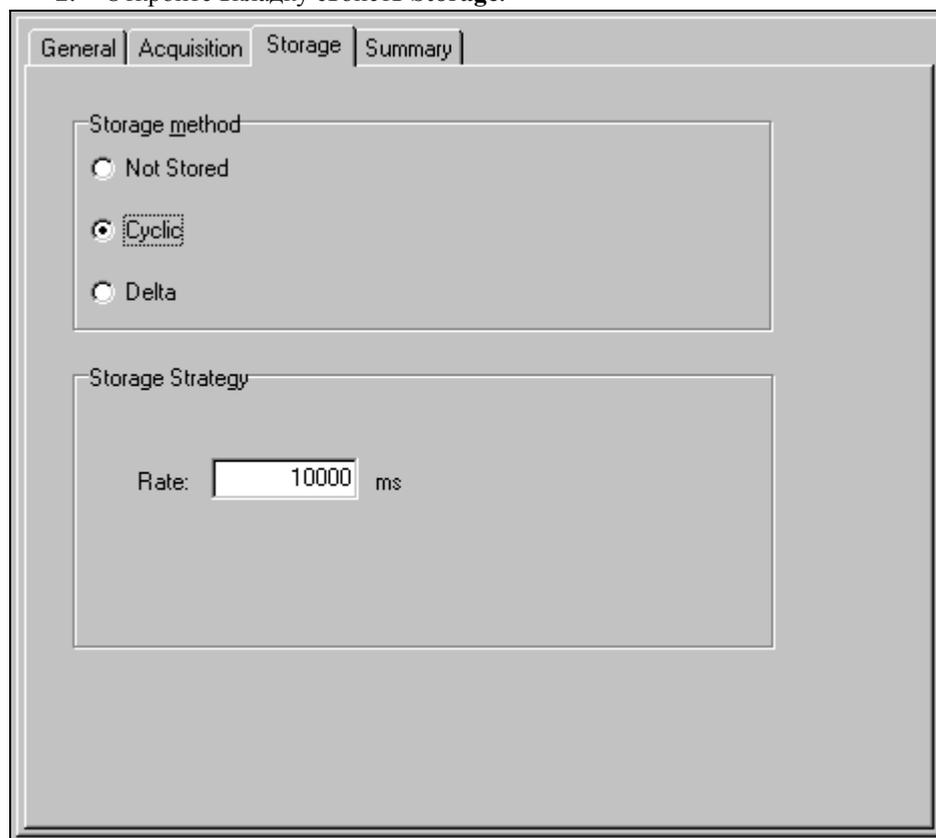
Свойства хранения данных настраиваются только для аналоговых тэгов. В общем случае, IndustrialSQL Server обеспечивает большую производительность и сжатие при использовании циклического метода хранения. Однако действуют исключения для:

- Данных, которые изменяются с непостоянным интервалом (то редко, то слишком часто).
- Данных, которые изменяются слишком медленно (не чаще, чем раз в 30 минут).

Подробнее о хранении данных см. в главе 4, "Управление системой хранения".

### ➤ Как отредактировать свойства хранения данных аналогового тэга:

1. Выберите в браузере аналоговый тэг.
2. Откройте вкладку свойств **Storage**.



3. В группе **Storage Method** выберите метод хранения значений тэга. **Not Stored**

Значения тэга не будут сохраняться системой хранения IndustrialSQL Server.

### Cyclic

Значения тэга сохраняются циклически. При использовании циклического метода данные сохраняются на основе временного интервала.

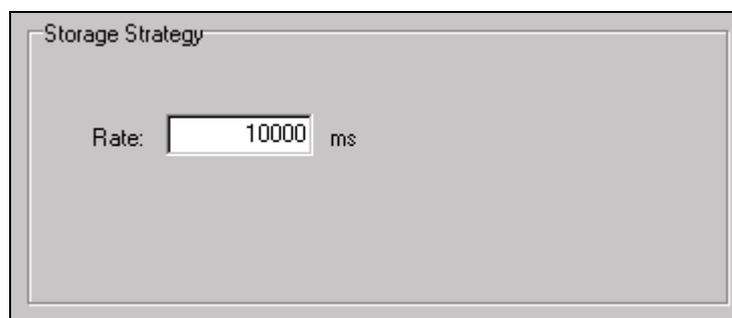
Импортированные тэги по умолчанию сохраняются циклическим методом. См. главу 3, "Интеграция InTouch".

### Delta

Значения тэга сохраняются методом исключения (дельта-хранение). При использовании дельта-метода данные сохраняются на основе изменений значения. Дельта-метод позволяет сохранять данные только при изменении значения тэга.

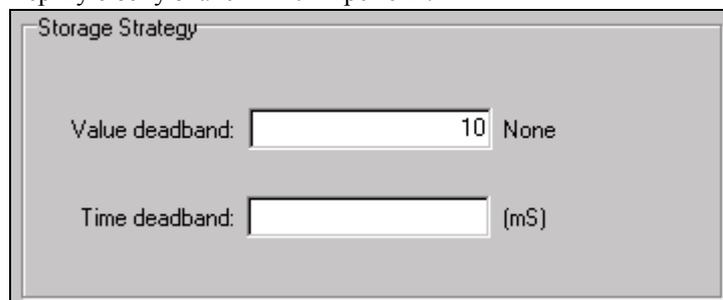
Параметры группы **Storage Strategy** изменяются, в зависимости от выбранного метода хранения.

4. Если выбран циклический метод, в поле **Rate** введите скорость хранения, или временной интервал, для тэга. Например:



The screenshot shows a dialog box titled "Storage Strategy". Inside, there is a label "Rate:" followed by a text input field containing the number "10000" and the unit "ms".

5. Если выбран метод исключения (дельта), настройте дополнительно мертвую зону значения или времени.



The screenshot shows a dialog box titled "Storage Strategy". It contains two input fields: "Value deadband:" with a text box containing "10" and the label "None" to its right, and "Time deadband:" with a text box and the label "(mS)" to its right.

### Value deadband

Мертвая зона значения для выборки данных определяет разрешение для хранения значений. Любые значения данных, которые изменяются меньше мертвой зоны, не будут сохраняться. Значение задается как процент от полного масштаба в единицах измерения.

### Time deadband

Мертвая зона времени для выборки данных определяет временное разрешение для хранения значений. Любые значения данных, которые изменяются в рамках временной мертвой зоны (в мс), не будут сохраняться.

6. Чтобы сохранить изменения, нажмите кнопку . Или в меню **File** выберите **Save**.
7. Чтобы отменить изменения, нажмите кнопку . Или в меню **File** выберите **Undo**.

## Редактирование свойств сводки аналогового тэга

Сводки — это операции агрегации, выполняемые автоматически с аналоговыми тэгами. Эти операции определены в системе управления событиями как операции сводок. Операция сводки обычно активизируется функцией планировщика, которая распознает наступление определенной даты/времени. Однако сводка может активизироваться в результате любого события.

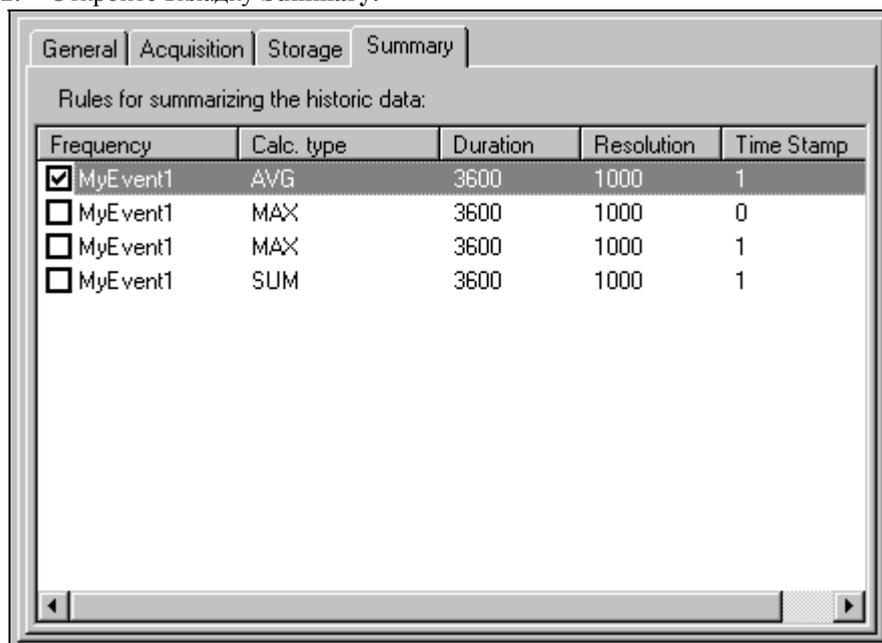
Результаты всех сводок хранятся в таблице *SummaryData* в базе данных *выполнения*. Система выполняет расчеты среднего, минимального, максимального и суммарного значений в зависимости от возникшего события.

См. главу 7, "Система управления событиями".

Если для аналогового тэга сконфигурирована операция сводки, вы можете просмотреть детали сводки и добавить или удалить тэг для сводки.

### ➤ Как отредактировать свойства сводки аналогового тэга:

1. Выберите в браузере аналоговый тэг.
2. Откройте вкладку **Summary**.



3. Столбец **Frequency** той операции сводки, в которую включен выбранный аналоговый тэг, будет помечен флажком.
4. Чтобы удалить тэг из операции сводки, нажмите на флажок, чтобы отключить его в столбце **Frequency**.
5. Чтобы добавить аналоговый тэг в операцию сводки, пометьте флажок в столбце **Frequency** у нужной операции.
6. Чтобы сохранить изменения, нажмите . Или в меню **File** выберите **Save**.

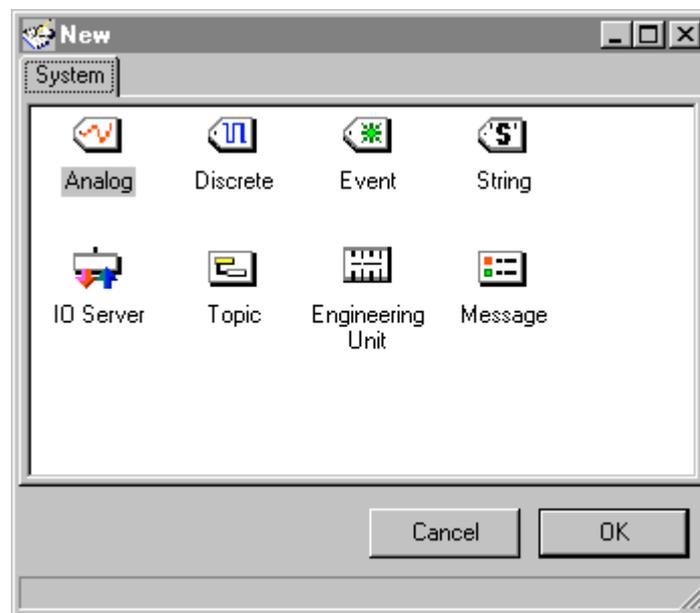
7. Чтобы отменить изменения, нажмите . Или в меню **File** выберите **Undo**.

## Добавление аналогового тэга в систему

Создать тэг можно только, если доступны сервер ввода-вывода и соответствующая тема.

### ➤ Как добавить аналоговый тэг:

1. В меню **File** выберите **New**. Или нажмите кнопку  на панели инструментов.
2. Появится диалоговое окно **New**.



---

**Совет.** Нажмите правой кнопкой мыши в этом окне, чтобы выбрать режим отображения пиктограмм.

---

2. Выберите пиктограмму **Analog** и нажмите **OK**.
3. Появится мастер **New Analog Tag**.
4. Введите уникальное имя аналогового тэга и нажмите **Next**. В имени тэга нельзя использовать пробелы, кавычки и другие специальные символы.
5. Мастер затем попросит ввести общие свойства, свойства сбора и хранения данных и свойства сводки для аналогового тэга.  
[Подробнее об общих свойствах см. в разделе "Редактирование общих свойств аналогового тэга" выше в этой главе.](#)  
[Подробнее о свойствах сбора данных см. в разделе "Редактирование свойств сбора данных аналогового тэга" выше в этой главе](#)  
[Подробнее о свойствах хранения данных см. в разделе "Редактирование свойств хранения данных аналогового тэга" выше в этой главе](#)
6. Закончив создание нового аналогового тэга, нажмите **Finish**.

## Настройка единиц измерения

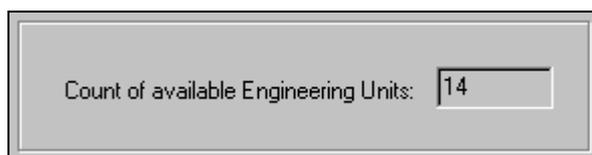
Единицей измерения для аналогового тэга может быть CPS (символов в секунду), Chars (символов), degrees (градусов) и т.п.

### Просмотр текущих единиц измерения сервера

Единицы измерения настраиваются для отдельного сервера. При создании или редактировании аналогового тэга можно использовать любую определенную в системе единицу измерения.

➤ **Как просмотреть текущий набор единиц измерения:**

1. В окне браузера разверните группу **System Object**, затем разверните группу **Servers** и выберите сервер, для которого вы хотите просмотреть текущий набор единиц измерения.
2. Нажмите на папку **Engineering Units**.
3. На панели свойств будет указано текущее количество единиц измерения.



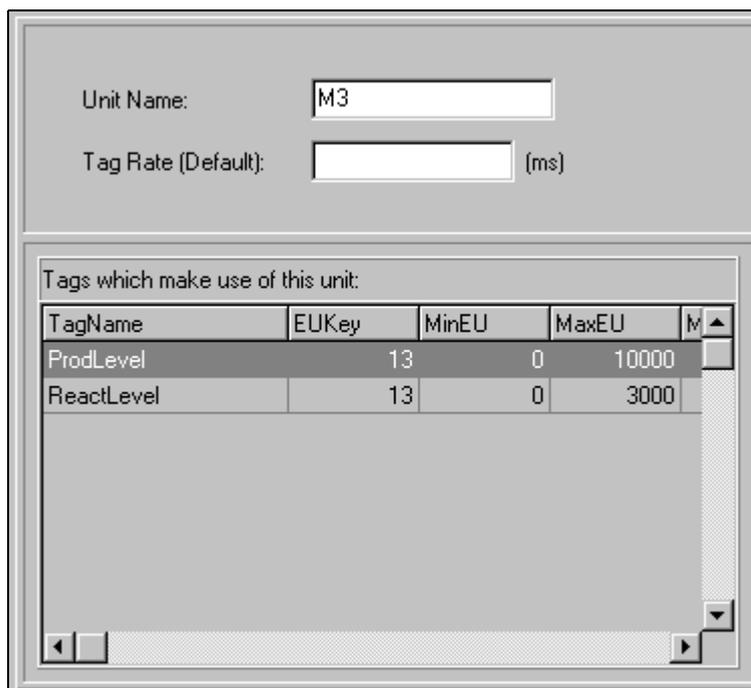
Все определенные единицы измерения отображаются на панели объектов браузера. Если выбрать единицу измерения в панели объектов, на панели свойств появится список использующих ее тэгов.

### Редактирование единицы измерения

Если внести изменения в единицу измерения, эти изменения будут действовать для всех использующих ее тэгов системы.

➤ **Как отредактировать единицу измерения:**

1. В окне браузера разверните группу **System Object**, затем разверните группу **Servers** и выберите сервер, для которого вы хотите отредактировать единицу измерения.
2. Нажмите на папку **Engineering Units**.
3. На панели объектов выберите нужную единицу измерения.



Unit Name:

Tag Rate (Default):  (ms)

Tags which make use of this unit:

TagName	EUKey	MinEU	MaxEU	M ▲
ProdLevel	13	0	10000	
ReactLevel	13	0	3000	

4. В поле **Unit Name** введите новое имя единицы измерения.
5. (необязательно) В поле **Tag Rate** укажите скорость по умолчанию в миллисекундах, с которой тэги будут циклически сохраняться, согласно единице измерения.

Введенное в этом поле значение, сохраняется в столбце *DefaultTagRate* таблицы *EngineeringUnit*. Хотя система не использует этот параметр скорости на базе единицы измерения, вы можете использовать ссылки на него в специализированных сценариях SQL. Значение, введенное в поле **Tag Rate**, не влияет на скорость сохранения тэга по умолчанию.

6. Чтобы сохранить изменения, нажмите кнопку . Или в меню **File** выберите **Save**.
7. Чтобы отменить изменения, нажмите кнопку . Или в меню **File** выберите **Undo**.

## Добавление единицы измерения

При добавлении единицы измерения можно настроить как существующие, так и новые тэги для использования новой единицы.

➤ **Как добавить единицу измерения:**

1. В меню **File** выберите **New**. Или нажмите кнопку  на панели инструментов.
2. Появится диалоговое окно **New**.



3.

---

**Совет.** Нажмите правой кнопкой мыши в этом окне, чтобы выбрать режим отображения пиктограмм.

---

2. Выберите пиктограмму **Engineering Unit** и нажмите **OK**.
3. Появится мастер **New Engineering Unit**.
4. Введите имя и (необязательно) скорость циклического сохранения для новой единицы измерения.  
*См. "Редактирование единицы измерения" выше в этой главе.*
5. Закончив создание новой единицы измерения, нажмите **Finish**.

## Настройка дискретных тэгов

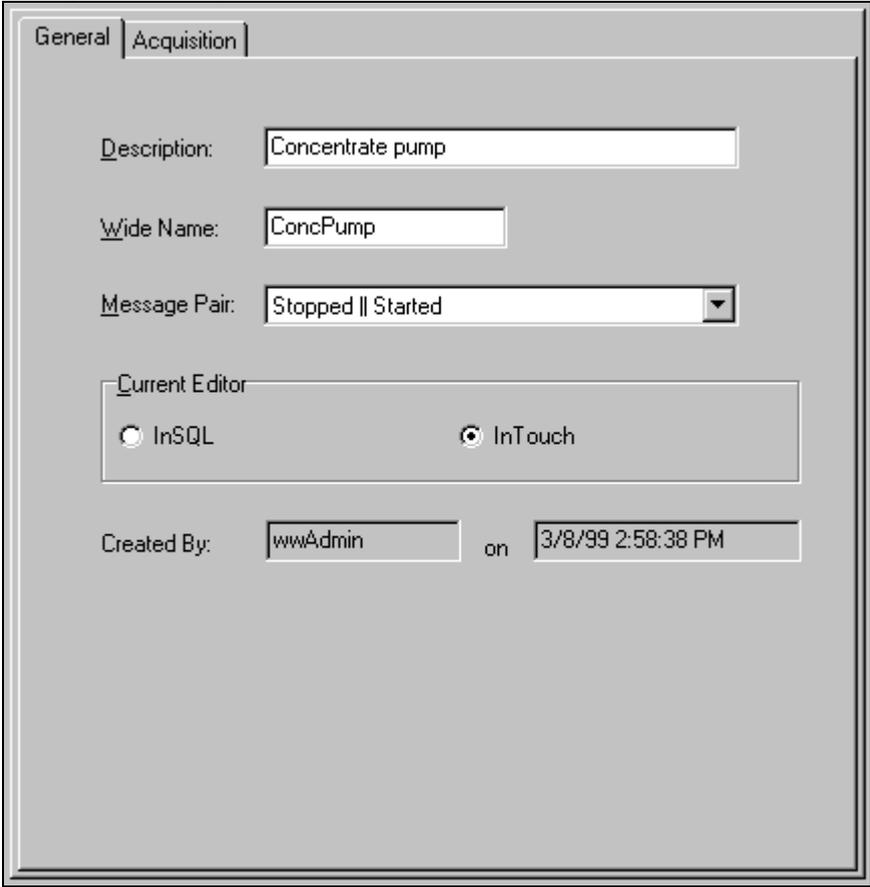
Для просмотра или редактирования конфигурации дискретного тэга нужно просто выбрать его в браузере. Информация тэга появится в соответствующей панели свойств. Можно настроить общие свойства, свойства сбора данных, а также можно добавить в систему новый дискретный тэг.

Информацию о настройке свойств сбора данных см. в разделе "Редактирование конфигурации сбора данных тэга" выше в этой главе.

### Редактирование общих свойств дискретного тэга

➤ **Как отредактировать общие свойства дискретного тэга:**

1. Выберите в браузере дискретный тэг.
2. Откройте вкладку свойств **General**.



The screenshot shows a configuration window with two tabs: "General" and "Acquisition". The "General" tab is selected. The window contains the following fields and controls:

- Description:** A text input field containing "Concentrate pump".
- Wide Name:** A text input field containing "ConcPump".
- Message Pair:** A dropdown menu showing "Stopped || Started".
- Current Editor:** A group box containing two radio buttons: "InSQL" (unselected) and "InTouch" (selected).
- Created By:** A text input field containing "wwAdmin" followed by "on" and a date/time field containing "3/8/99 2:58:38 PM".

3. В поле **Description** введите описание тэга.

4. В поле **Wide Name** введите версию тэга в виде допустимого для SQL Server имени столбца.

В широких таблицах (*WideTableDictionary*, *AnalogWideHistory* и *DiscreteWideHistory*) один или несколько имен тэгов используются как имена столбцов. Если из InTouch импортировано имя тэга, содержащее недопустимые символы, его следует переименовать согласно правилам для идентификаторов Microsoft SQL Server. См. документацию по Microsoft SQL Server.

5. В списке **Message Pair** выберите пару сообщений, которая будет связана с состоянием ЛОЖЬ/ИСТИНА дискретного тэга.  
См. "Настройка пар сообщений" далее в этой главе.
6. В группе **Current Editor** укажите, где будет редактироваться тэг — в IndustrialSQL Server или в InTouch.

Параметр Current Editor определяет, в какой среде будет редактироваться тэг, либо в InSQL Configure или в InTouch (через функцию импорта тэгов). Тэги, импортированные из InTouch, изначально настроены на InTouch как среду редактирования. Если с тэгом производятся изменения в InSQL Configure, то редактор тэга меняется на InSQL Configure. При повторном импортировании все изменения, сделанные с помощью InSQL Configure, сохраняются. Можно вручную сохранять InTouch в качестве редактора для повторного импортирования; однако при этом любые изменения, сделанные в тэге с помощью InSQL Configure, будут потеряны во время повторного импортирования.

7. Чтобы сохранить изменения, нажмите кнопку . Или в меню **File** выберите **Save**.
8. Чтобы отменить изменения, нажмите кнопку . Или в меню **File** выберите **Undo**.

Следующие параметры вкладки **General** нельзя редактировать для дискретного тэга.

#### **Created By**

Имя пользователя или приложения, создавшего тэг. Генерируется автоматически при наполнении базы данных.

#### **On**

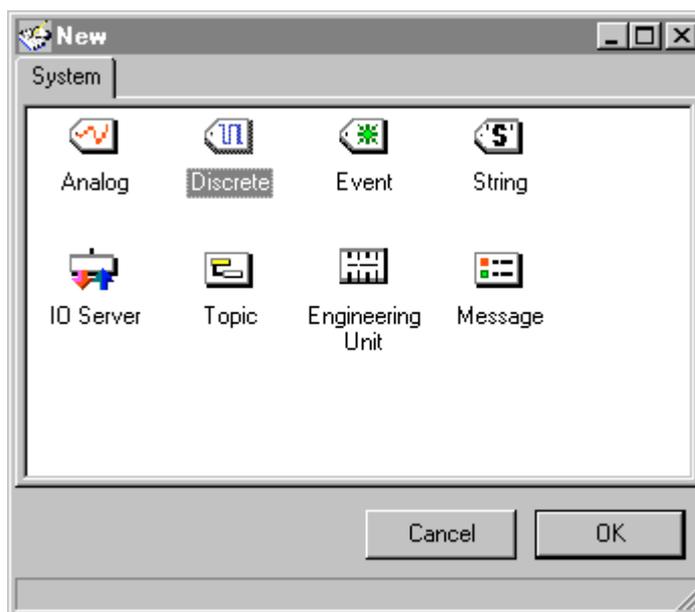
Дата создания тэга. Генерируется автоматически при наполнении базы данных.

## Добавление в систему дискретного тэга

Создать тэг можно только, если доступны сервер ввода-вывода и соответствующая тема.

### ➤ Как добавить дискретный тэг:

1. В меню **File** выберите **New**. Или нажмите кнопку  на панели инструментов.
2. Появится диалоговое окно **New**.



3.

---

**Совет.** Нажмите правой кнопкой мыши в этом окне, чтобы выбрать режим отображения пиктограмм.

---

2. Выберите пиктограмму **Discrete** и нажмите **OK**.
3. Появится мастер **New Discrete Tag**.
4. Введите уникальное имя дискретного тэга и нажмите **Next**. В имени тэга нельзя использовать пробелы, кавычки и другие специальные символы.
5. Мастер затем попросит ввести общие свойства, свойства сбора и хранения данных и свойства сводки для нового тэга.  
*Подробнее об общих свойствах см. в разделе "Редактирование общих свойств дискретного тэга" выше в этой главе.*  
*Подробнее о свойствах сбора данных см. в разделе "Редактирование свойств сбора данных тэга" выше в этой главе.*
6. Закончив создание нового дискретного тэга, нажмите **Finish**.

## Настройка пар сообщений

Парой сообщений называется два сообщения, связанных с состоянием ЛОЖЬ или ИСТИНА дискретного тэга. Дискретный тэга со значением 0 имеет состояние ЛОЖЬ или ВЫКЛЮЧЕН. Дискретный тэга со значением 1 имеет состояние ИСТИНА или ВКЛЮЧЕН.

### Просмотр текущих пар сообщений сервера

Пары сообщений настраиваются для отдельного сервера. При создании или редактировании дискретного тэга можно использовать любую определенную в системе пару сообщений.

➤ **Как просмотреть текущий набор пар сообщений:**

1. В окне браузера разверните группу **System Object**, затем разверните группу **Servers** и выберите сервер, для которого вы хотите просмотреть текущий набор пар сообщений.
2. Откройте вкладку **Messages**.
3. На панели свойств будет указано текущее количество пар сообщений.



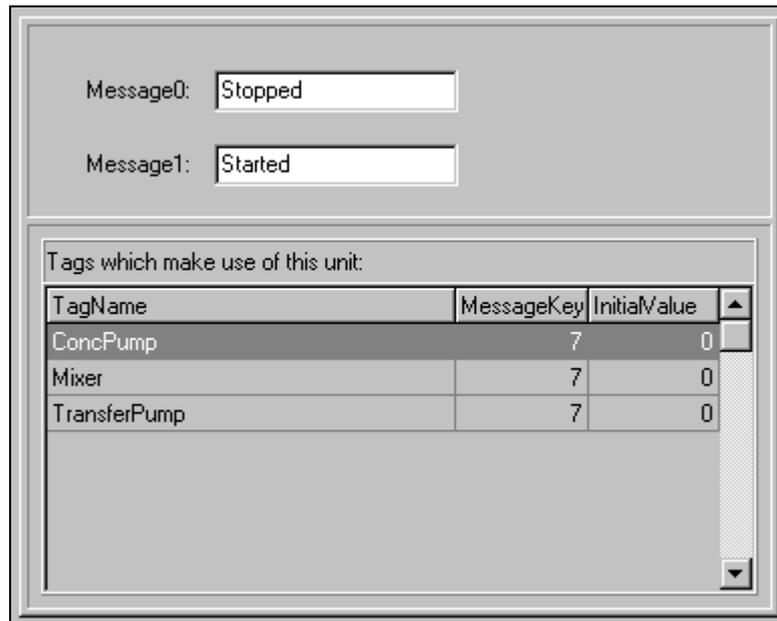
Все определенные пары сообщений отображаются на панели объектов браузера. Если выбрать пару сообщений в панели объектов, на панели свойств появится список использующих ее тэгов.

### Редактирование пары сообщений

Если внести изменения в пару сообщений, эти изменения будут действовать для всех использующих ее тэгов системы.

➤ **Как отредактировать пару сообщений:**

1. В окне браузера выберите сервер, для которого вы хотите отредактировать пару сообщений.
2. Откройте вкладку **Messages**.
3. Выберите нужную пару сообщений.



Message0:

Message1:

Tags which make use of this unit:

TagName	MessageKey	InitialValue
ConcPump	7	0
Mixer	7	0
TransferPump	7	0

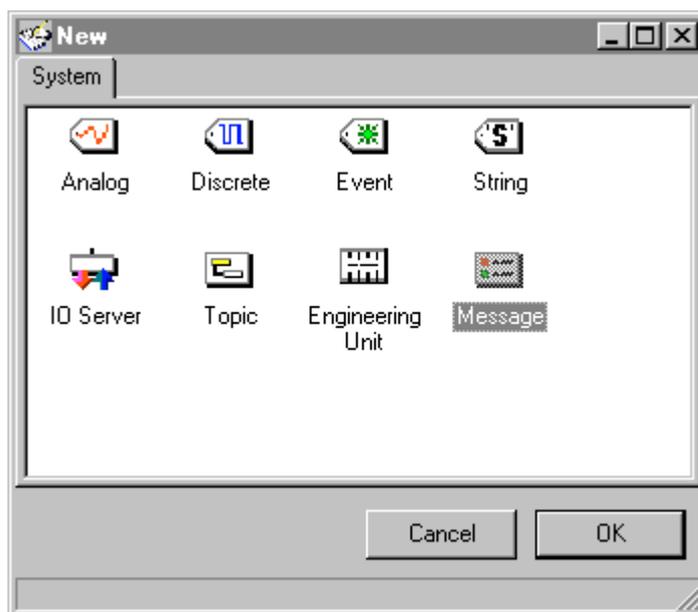
4. В поле **Message0** введите сообщение, связанное с состоянием ЛОЖЬ дискретного тэга. Дискретный тэг со значением 0 находится в состоянии ЛОЖЬ.
5. В сообщении нельзя использовать пробелы и другие специальные символы.
6. В поле **Message1** введите сообщение, связанное с состоянием ИСТИНА дискретного тэга. Дискретный тэг со значением 1 находится в состоянии ИСТИНА.
7. Чтобы сохранить изменения, нажмите кнопку  Или в меню **File** выберите **Save**.
8. Чтобы отменить изменения, нажмите кнопку  Или в меню **File** выберите **Undo**.

## Добавление пары сообщений

При добавлении пары сообщений можно настроить как существующие, так и новые тэги для использования новой пары.

➤ **Как добавить пару сообщений:**

1. В меню **File** выберите **New**. Или нажмите кнопку  на панели инструментов.
2. Появится диалоговое окно **New**.



---

**Совет.** Нажмите правой кнопкой мыши в этом окне, чтобы выбрать режим отображения пиктограмм.

---

2. Выберите пиктограмму **Message** и нажмите **OK**.
3. Появится мастер **New Message**.
4. Мастер затем попросит указать сообщения для состояний ЛОЖЬ и ИСТИНА дискретного тэга. Дискретный тэг со значением 0 имеет состояние ЛОЖЬ или ВЫКЛЮЧЕН. Дискретный тэг со значением 1 имеет состояние ИСТИНА или ВКЛЮЧЕН.  
[См. "Редактирование пары сообщений" выше в этой главе.](#)
5. Закончив создание новой пары сообщений, нажмите **Finish**.

## Настройка текстовых тэгов

Для просмотра или редактирования конфигурации текстового тэга нужно просто выбрать его в браузере. Информация тэга появится в соответствующей панели свойств. Можно настроить общие свойства, свойства сбора данных, а также можно добавить в систему новый текстовый тэг.

Информацию о настройке свойств сбора данных см. в разделе "Редактирование конфигурации сбора данных тэга" выше в этой главе.

### Редактирование общих свойств текстового тэга

➤ **Как отредактировать общие свойства текстового тэга:**

1. Выберите в браузере текстовый тэг.
2. Откройте вкладку свойств **General**.



The screenshot shows a configuration window with two tabs: "General" and "Acquisition". The "General" tab is selected. The window contains the following fields and controls:

- Description:** A text input field containing "Operator".
- Wide Name:** A text input field containing "\w\$Operator".
- Max Length:** A text input field containing "0".
- Current Editor:** A group box containing two radio buttons: "InSQL" (unselected) and "InTouch" (selected).
- Created By:** A text input field containing "wwAdmin" followed by "on" and a date/time field containing "9/15/97 4:56:13 AM".

3. В поле **Description** введите описание тэга.
4. В поле **Max Length** укажите максимальную длину тэга в байтах.

5. В поле **Wide Name** введите версию тэга в виде допустимого для SQL Server имени столбца.

В широких таблицах (*WideTableDictionary*, *AnalogWideHistory* и *DiscreteWideHistory*) один или несколько имен тэгов используются как имена столбцов. Если из InTouch импортировано имя тэга, содержащее недопустимые символы, его следует переименовать согласно правилам для идентификаторов Microsoft SQL Server. См. документацию по Microsoft SQL Server.

6. В группе **Current Editor** укажите, где будет редактироваться тэг — в IndustrialSQL Server или в InTouch.

Параметр Current Editor определяет, в какой среде будет редактироваться тэг, либо в InSQL Configure или в InTouch (через функцию импорта тэгов). Тэги, импортированные из InTouch, изначально настроены на InTouch как среду редактирования. Если с тэгом производятся изменения в InSQL Configure, то редактор тэга меняется на InSQL Configure. При повторном импортировании все изменения, сделанные с помощью InSQL Configure, сохраняются. Можно вручную сохранять InTouch в качестве редактора для повторного импортирования; однако любые изменения, сделанные в тэге с помощью InSQL Configure, будут потеряны во время повторного импортирования.

7. Чтобы сохранить изменения, нажмите кнопку . Или в меню **File** выберите **Save**.

8. Чтобы отменить изменения, нажмите кнопку . Или в меню **File** выберите **Undo**.

Следующие параметры вкладки **General** нельзя редактировать для текстового тэга.

#### **Created By**

Имя пользователя или приложения, создавшего тэг. Генерируется автоматически при наполнении базы данных.

#### **On**

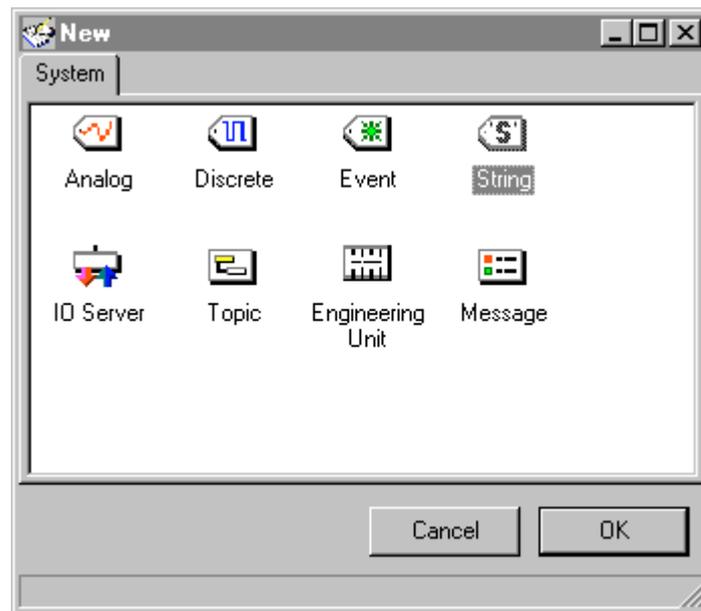
Дата создания тэга. Генерируется автоматически при наполнении базы данных.

## Добавление в систему текстового тэга

Создать тэг можно только, если доступны сервер ввода-вывода и соответствующая тема.

### ➤ Как добавить текстовый тэг:

1. В меню **File** выберите **New**. Или нажмите кнопку  на панели инструментов.
2. Появится диалоговое окно **New**.



---

**Совет.** Нажмите правой кнопкой мыши в этом окне, чтобы выбрать режим отображения пиктограмм.

---

2. Выберите пиктограмму **String** и нажмите **OK**.
3. Появится мастер **New String Tag**.
4. Введите уникальное имя текстового тэга и нажмите **Next**. В имени тэга нельзя использовать пробелы, кавычки и другие специальные символы.
5. Мастер затем попросит ввести общие свойства и свойства сбора данных для нового тэга.  
*Подробнее об общих свойствах см. в разделе "Редактирование общих свойств дискретного тэга" выше в этой главе.*  
*Подробнее о свойствах сбора данных см. в разделе "Редактирование свойств сбора данных тэга" выше в этой главе.*
6. Закончив создание нового текстового тэга, нажмите **Finish**.

## Настройка тэгов событий

*Тэгом события* называется определение события в системе. Например, если нужно фиксировать как событие ситуацию, когда температура емкости поднимается до 100 градусов, можно определить для него тэг под именем "TankAt100". Тэги событий отличаются от всех других типов тэгов в IndustrialSQL Server (аналоговых, дискретных и текстовых). Если эти типы тэгов представляют собой определения сохраняемых переменных, то тэг события является определением ссылки на определенное событие, которое требуется фиксировать, и действия, требуемое в ответ на возникновение события. Определение тэга события служит в системе в качестве ссылки на всю эту информацию о событии.

Процедура определения тэга события включает себя:

- Указание общей информации тэга события, например, имя и описание тэга.
- Указание события, которое будет распознаваться системой.
- Указание действия, которое будет выполняться при возникновении события.

Подробнее о тэгах событий см. в главе 7, "Система управления событиями" этого руководства.

## Настройка информации о местах хранения

Местом хранения называется место, где IndustrialSQL Server хранит производственные данные. Каждое место хранения может включать в себя одну или несколько следующих областей хранения: циклическая, постоянная, буферная и альтернативная. С помощью InSQL Configure можно просмотреть и отредактировать пути к этим областям. Эти пути изначально задаются при установке IndustrialSQL Server.

Для редактирования путей к областям хранения необходимо иметь привилегии администратора.

---

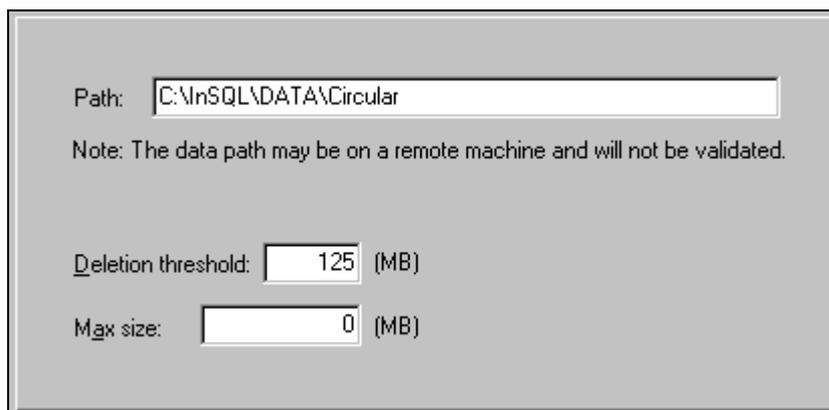
**Примечание.** Пути к областям хранения задаются относительно компьютера, на котором работает IndustrialSQL Server. Если запускать InSQL Configure не на том компьютере, где установлен IndustrialSQL Server, пути могут отличаться.

---

См. также главу 4, "Управление системой хранения".

### ➤ Как отредактировать места хранения:

1. В окне браузера разверните группу **System Object**, затем разверните группу **Servers** и выберите сервер, для которого вы хотите отредактировать места хранения.
2. Откройте вкладку **Storage Locations**.
3. Выберите нужную область хранения.
4. На панели свойств будет показана информация об этой области хранения.



Path:

Note: The data path may be on a remote machine and will not be validated.

Deletion threshold:  (MB)

Max size:  (MB)

5. В поле **Path** введите путь к области хранения.
6. В поле **Deletion Threshold** укажите порог дискового пространства, при достижении которого система начнет удалять старые данные.
7. В поле **Max Size** укажите максимальный объем данных, который может храниться в данной области.  
Если здесь указать 0, то данная область хранения будет использовать все свободное дисковое пространство.
8. Чтобы сохранить изменения, нажмите кнопку  Или в меню **File** выберите **Save**.
9. Чтобы отменить изменения, нажмите кнопку  Или в меню **File** выберите **Undo**.

За исключением циклической области хранения, изменения всех других путей активизируются динамически. Только изменение пути циклической области требует повторной инициализации системы.

Кроме того, если изменяется путь по умолчанию, необходимо вручную создать соответствующие каталоги. InSQL Configure не проверяет наличие каталогов, указанных в пути.

## Настройка серверов ввода-вывода

Сервер ввода-вывода — это приложение, предоставляющее данные клиенту через определенный протокол. Существует три типа серверов ввода-вывода, используемых с IndustrialSQL Server:

- Внешние серверы ввода-вывода, передающие данные в IndustrialSQL Server по протоколу DDE или SuiteLink. Примерами служат серверы ввода-вывода Wonderware, серверы ввода-вывода сторонних производителей и созданные пользователями.
- Системный сервер ввода-вывода, который представляет собой внутренний драйвер, передающий данные из внутренних переменных системы.
- Сервер ввода-вывода InSQL, представляющий собой интерфейс для доступа клиентов к данным реального времени, хранящимся в IndustrialSQL Server.

Серверы ввода-вывода и соответствующие темы могут быть импортированы из InTouch или добавлены вручную с помощью InSQL Configure.

См. также главу 10, "Настройка серверов ввода-вывода".

## Повторная инициализация

Повторная инициализация — это перезапуск одного или нескольких процессов в системе IndustrialSQL Server. Повторная инициализация может потребоваться после внесения изменений в базу данных *выполнения* для их активизации. При изменении настроек с помощью InSQL Configure эта программа может выдать запрос на повторную инициализацию после операции импорта или после выхода из программы.

---

**Примечание.** Повторная инициализация всегда требуется после операций импорта, включая дельта-реимпортирование.

---

Повторную инициализацию IndustrialSQL Server можно выполнить вручную, выбрав команду **Reinitialize Server** в меню **Tools**. Появится запрос на подтверждение повторной инициализации. Для повторной инициализации сервера необходимо иметь привилегии администратора.

В следующей таблице перечислены изменения конфигурации, требующие повторной инициализации.

Объект	Операция	Поле	Последствие
Место хранения Аналоговый тэг	Обновление	Циклическое	Перезапуск
	Вставка	-	Перезапуск
Дискретный тэг	Удаление	-	Перезапуск
	Обновление	Topic, IO Server, Acquisition Type, Storage Type, Storage Rate, Item Name, Min Raw, Max Raw, Raw Type, Integer Size, Signed Integer Min EU, Max EU, Scaling	Перезапуск
	Вставка	Любое поле сводки	Нет <sup>1</sup>
	Удаление	Любое другое поле	Нет
Текстовый тэг	Обновление	Topic, IO Server, Item Name, Message Любое другое поле	Перезапуск
	Вставка	-	Нет
	Удаление	-	Перезапуск
Текстовый тэг	Обновление	-	Нет
	Вставка	-	Перезапуск
Текстовый тэг	Удаление	-	Перезапуск
	Обновление	-	Нет

*(продолжение)*

Объект	Операция	Поле	Последствие
Тэг события	Вставка	-	Нет <sup>1</sup>
	Удаление	-	Нет <sup>1</sup>
	Обновление	-	Нет <sup>1</sup>
Сервер ввода-вывода	Вставка	-	Перезапуск
	Удаление	-	Перезапуск
	Обновление	Любое поле	Перезапуск
Пара сообщений	Вставка	-	Нет
	Удаление	-	Нет
	Обновление	Любое поле	Нет
Единица измерения	Вставка	-	Нет
	Удаление	-	Нет
	Обновление	Любое поле	Нет

*Примечания*

<sup>1</sup> Система управления событиями автоматически обрабатывает изменения в конфигурации тэгов событий.

---

**Примечание.** Изменения атрибутов *TimeDeadband* или *ValueDeadband* для дельта-хранения аналоговых тэгов обрабатываются автоматически без необходимости перезапуска системы хранения.

---

## Хранимые процедуры для повторной инициализации

Повторная инициализация зависит от следующих хранимых процедур:

- `ww_InSQLStatusSet`
- `ww_InSQLStatusSelect`

Хранимая процедура `ww_InSQLStatusSelect` используется программами InSQL Configure и InSQL Control для определения статуса изменения базы данных. Статус изменения определяется по значению флага статуса базы данных (*DbStatus*) и типу изменения (*DbModAcquisition*, *DbModStorage*, *DbModServer*, *DbModAll*) из таблицы *StorageNode*.

Когда производится изменение конфигурации базы данных *выполнения* с помощью InSQL Configure, атрибут *DbStatus* в таблице *StorageNode* принимает значение больше 0, означая, что изменения значительные и требуется повторная инициализация. InSQL Configure использует хранимую процедуру `ww_InSQLStatusSet` для установки флагу статуса, *DbStatus*, значения больше 0, когда изменение в базе данных требует обработки сервером.

Программа InSQL Control, заметив значительное изменение, судя по значению флага *DbStatus*, вносит необходимые изменения, при необходимости повторно инициализирует IndustrialSQL Server, а затем снова устанавливает на флаг *DbStatus* значение 0, означающий, что повторная инициализация выполнена.

Подробнее о хранимых процедурах см. в *Справочном руководстве IndustrialSQL Server*.

## Отслеживание изменений

IndustrialSQL Server поддерживает отслеживание изменений в тэгах базы данных. Если на предприятии существуют правила по отслеживанию изменений, можно настроить IndustrialSQL Server на отслеживание изменений. Например, можно вести учет всех изменений в конфигурации аналоговых тэгов.

Изменения могут отслеживаться в следующих типах столбцов:

- Real (действительный, включая с плавающей точкой)
- Char (символьный любой длины)
- Integer (целый, включая *tinyint*, *smallint*, *int*)
- Decimal (любого десятичного формата)
- Binary (двоичный)
- Bit (битовый)
- DateTime (включая *datetime*, *smalldatetime*)
- Text (текстовый)

## Таблицы отслеживания изменений

Все объекты, для которых возможно отслеживание изменений, хранятся в таблице *InSQLSysObjects*. Когда в столбец базы данных вносятся изменения, строка добавляется в таблицу *ModLog* и в таблицу, соответствующую типу столбца измененного значения.

Подробнее о таблицах отслеживания изменений см. в *Справочном руководстве IndustrialSQL Server*.

## Особенности и ограничения отслеживания изменений

Отслеживание изменений носит общесистемный характер. Эту функцию нельзя включить или отключить на уровне одной таблицы. Включение функции отслеживания изменений приводит к снижению производительности IndustrialSQL Server, когда в систему вносятся изменения. Это вызвано потребностью в дополнительной обработке и дополнительном пространстве для учета изменений. Однако в среде выполнения снижения производительности не наблюдается.

---

**Примечание.** В этой версии отслеживание изменений ведется только в следующих таблицах: *Tag*, *AnalogTag*, *DiscreteTag*, *StringTag*, *EventTag*, *ComplexTag*, *Topic*, *IOServer* и *EngineeringUnit*.

---

Каждое изменение столбца приводит к вставке строки в таблицу *ModLog*. Например, если добавить в систему аналоговый тэг, будет изменено в сумме 27 столбцов (с учетом таблиц *Tag* и *AnalogTag*), и в таблицу *ModLog* будет

добавлено 27 строк. Если изменить один атрибут тэга (столбец), в базе будет произведено два изменения: изменение в таблицах тэгов и вставка строки в таблицы отслеживания.

Информация таблиц отслеживания изменений хранится в устройствах данных (файлах данных) базы Microsoft SQL Server. Когда отслеживание изменений включено, объем данных в этих устройствах значительно увеличивается.

Поэтому рекомендуется:

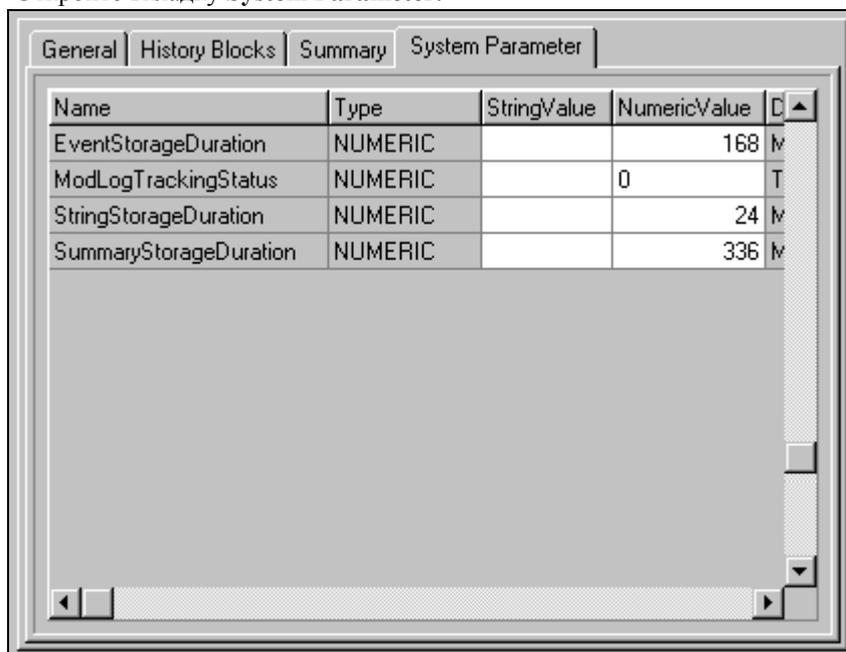
- Отключить отслеживание изменений перед первой операцией импорта.
- Включить отслеживание изменений перед повторным импортированием или при небольшом объеме изменений.

Отслеживание изменений не поддерживает обработку пакетного типа. Если основной символ (*TagName*) изменяется для нескольких тэгов (строк) одновременно, изменения во всем пакете не будут отслеживаться.

## Включение и выключение отслеживания изменений

### ➤ Как включить и выключить отслеживание изменений:

1. В окне браузера разверните группу **System Object**, затем разверните группу **Servers** и выберите сервер, для которого вы хотите включить или выключить отслеживание изменений.
2. Откройте вкладку **System Parameter**.



Name	Type	StringValue	NumericValue	
EventStorageDuration	NUMERIC		168	M
ModLogTrackingStatus	NUMERIC		0	T
StringStorageDuration	NUMERIC		24	M
SummaryStorageDuration	NUMERIC		336	M

3. Чтобы включить отслеживание изменений, введите 1 в столбец **NumericValue** для **ModLogTrackingStatus**.
4. Чтобы выключить отслеживание изменений, введите 0 в столбец **NumericValue** для **ModLogTrackingStatus**.
5. Чтобы сохранить изменения, нажмите кнопку . Или в меню **File** выберите **Save**.

6. Чтобы отменить изменения, нажмите кнопку . Или в меню **File** выберите **Undo**.

## Управление дисковым пространством

IndustrialSQL Server периодически удаляет старые данные из таблиц *StringHistory*, *EventHistory* и *SummaryHistory*. Срок хранения архивов текстовых, сводных данных и событий задается в таблице *NamedSystemParameter*.

По умолчанию действует следующий срок хранения:

Архив	Срок хранения
Текстовые данные	1 день (24 часа)
Данные событий	7 дней (168 часов)
Сводные данные	14 дней (336 часов)

- **Как изменить срок хранения архивов текстовых, сводных данных и событий:**
1. В окне браузера разверните группу **System Object**, затем разверните группу **Servers** и выберите сервер, для которого вы хотите изменить срок хранения.
  2. Откройте вкладку **System Parameter**.
  3. Введите новый срок хранения для требуемого типа архивных данных.
  4. Чтобы сохранить изменения, нажмите кнопку . Или в меню **File** выберите **Save**.
  5. Чтобы отменить изменения, нажмите кнопку . Или в меню **File** выберите **Undo**.

## Г Л А В А 3

# Интеграция InTouch

В этой главе описана методика интеграции IndustrialSQL Server в систему InTouch.

Подробнее об использовании InTouch см. *Руководство пользователя InTouch*.

## Содержание

- [Импортирование словаря тэгов InTouch](#)
- [IndustrialSQL как поставщик архива для InTouch](#)
- [Элементы ActiveX системы IndustrialSQL Server](#)
- [Использование InTouch SQL Access Manager](#)

## Импортирование словаря тэгов InTouch

Словарь тэгов InTouch используется для настройки большинства систем IndustrialSQL Server. Использование функции импорта InSQL Configure позволяет легко импортировать определения из словарей тэгов нескольких узлов InTouch (одно приложение на узел) в базу данных *Runtime* IndustrialSQL Server. Импортирование определений тэгов избавляет от необходимости вводить вручную информацию сервера ввода-вывода и тэгов как для InTouch, так и для системы IndustrialSQL Server. Когда словарь тэгов импортирован, данные, определенные для приложения InTouch, сохраняются в базе данных *Runtime*.

Подробнее см. главу 2, "InSQL Configure", в этом руководстве администратора.

Когда словарь тэгов импортируется из InTouch, он автоматически направляется в соответствующие таблицы базы данных *Runtime* IndustrialSQL Server. Пользовательские тэги InTouch поддерживаются в общих именованных пространствах. Когда данные сохранены в базе данных *Runtime*, ими можно манипулировать при помощи любого метода выборки данных SQL, поддерживаемого системой IndustrialSQL Server. К этим данным могут быть применены функции, характерные для IndustrialSQL Server, такие как установка разрешения запроса, благодаря чему данные могут быть получены от любого клиента Wonderware, включая InTouch. Подробнее см. *Руководство пользователя InTouch*.

---

**Примечание.** Импортировать можно словари данных из InTouch версии 6.0 и выше. Однако для дельта-импортирования следует импортировать словарь тэгов InTouch версии 7.1. Право импортирования имеет только пользователь с уровнем "wwAdmin" или "wwdbo".

---

## Об импортировании

Использование функции импорта тэгов программы InSQL Configure позволяет импортировать темы и другие данные настройки из одного и нескольких узлов InTouch. Существует несколько факторов, влияющих на эффективность импорта тэгов из множества узлов InTouch в базу данных IndustrialSQL Server.

Хотя можно импортировать данные из множества узлов InTouch, с одного узла может быть импортировано только одно приложение. Вместе с тем, если приложение удалено с IndustrialSQL Server, можно загрузить другое приложение с того же узла InTouch.

При удалении приложения с IndustrialSQL Server будут удалены вся информация о тэгах и все сохраненные данные для этих тэгов. Если производится повторное импортирование, будут сохранены существующие определения в IndustrialSQL Server.

## Определение порядка импортирования

Для импортирования из нескольких узлов InTouch можно использовать следующие сценарии:

1. Существует несколько узлов InTouch, получающих значения от одного и того же тэга ввода-вывода, который в свою очередь получает значения от одного и того же источника на промышленном устройстве.
2. Существует узел InTouch, получающий значения с тэга ввода-вывода другого узла InTouch, который получает значения от источника, расположенного на промышленном устройстве.
3. Все узлы InTouch получают значения от тэгов ввода-вывода с выделенного узла InTouch, который получает значения данных из источника на промышленном устройстве. (В данном случае, выделенный узел InTouch настроен на работу в качестве "сервера тэгов").

При импорте словарей тэгов со множества узлов InTouch естественным желанием пользователя будет устранение повторного импорта тэгов. Для максимальной эффективности импортирования данных сначала следует импортировать узел InTouch, функционирующий в качестве сервера тэгов или содержащий наибольшее количество тэгов, имеющих прямой доступ к источникам данных.

При импорте множества узлов следует импортировать узел с большим количеством тэгов с прямым доступом перед тем, как импортировать узел с меньшим количеством таких тэгов.

## Дублирование тэгов

Если при импорте тэгов встречается имя тэга, уже существующее в базе данных IndustrialSQL Server, то к его началу или концу будет добавлен символ, унифицирующий его в IndustrialSQL Server. Во время импорта можно выбрать:

- Какой символ будет добавлен.
- Куда будет добавлен символ — в начало или в конец имени тэга.
- Следует ли добавлять символ во все имена тэгов этого узла, несмотря на то, дублируются они или нет, или же добавлять его только в дублируемые тэги.

Если при добавлении символа во время процесса импортирования будет создан дублируемый тэг, функция импорта не будет импортировать этот тэг. Например, вы выбираете букву "B" в качестве добавляемого символа для конкретного узла и импортируете тэг с именем "TestTag". Но тэг с именем "BTestTag" уже присутствует в IndustrialSQL Server. Функция импорта не будет импортировать потенциально дублируемые тэги. Список таких тэгов будет выведен после окончания импорта. Для обхода этой проблемы измените имена этих тэгов в InTouch во избежание дублирования и попробуйте импортировать их еще раз.

## Дублирование адресов тэгов

Если функция импорта встречает дублированный адрес (состоящий из имен приложения, темы и элемента) у одного или нескольких тэгов, будет импортирован только первый тэг. Все остальные будут пропущены.

## Редактирование имен компьютеров

Во время импорта будет предложено указать имя компьютера InTouch и путь к приложению InTouch. При первой операции импортирования словаря тэгов из узла InTouch по умолчанию будут показаны текущие имя компьютера и путь к приложению InTouch.

Обычно эти стандартные значения изменять не следует. Тем не менее, если не получается физически установить связь с приложением InTouch (например, выключена сеть), можно скопировать приложение на другой компьютер и импортировать оттуда. В этом случае следует изменить стандартное имя компьютера и путь к приложению для указания места нахождения оригинального приложения.

Во время повторного импортирования невозможно изменить имя компьютера InTouch и путь к приложению InTouch. Однако эту информацию можно отредактировать в таблице *InTouchNode* с помощью Query Analyzer (или иного инструмента запросов) до повторного импортирования. После повторного импортирования будет использоваться новая информация.

## Дельта-реимпортирование

"Дельта-реимпортирование" — это повторное импортирование только тех тэгов, которые изменились в конкретном узле InTouch после предыдущей операции импортирования. Дельта-реимпортирование позволяет быстрее производить повторное импортирование, поскольку при этом не обновляются все тэги в базе данных. Тем не менее, дельта-реимпортирование не дает абсолютной свободы, характерной для полной операции повторного импортирования. Нельзя импортировать *поднабор* измененных тэгов или изменить скорость циклического сохранения. Для использования этих возможностей следует произвести полное повторное импортирование. Например, если при первом импортировании была импортирована Тема А, а Тема Б - нет, то для добавления Темы Б в базу данных IndustrialSQL Server потребуется полное повторное импортирование.

Любые унифицирующие настройки и параметры циклического сохранения будут сохранены для дельта-реимпортирования.

---

**Примечание.** Дельта-реимпортирование поддерживается только в InTouch 7.1.

---

## Импортирование конфигурации сервера ввода-вывода

При импортировании конфигурации удаленных серверов ввода-вывода, использующих DDE, следует сначала настроить общие ресурсы этих серверов ввода-вывода и затем импортировать словарь тэгов в IndustrialSQL Server. Для локальных серверов DDE настраивать общие ресурсы необязательно.

Подробнее см. в *Руководстве администратора системы FactorySuite*.

При импортировании конфигурации сервера ввода-вывода из InTouch стандартный протокол для всех серверов ввода-вывода будет установлен на SuiteLink. После импортирования сервера(ов) ввода-вывода DDE потребуется запустить InSQL Configure, чтобы изменить протокол обратно на DDE.

Подробнее см. в главе 2, "InSQL Configure", в этом руководстве администратора.

## Резервирование базы данных Runtime

Прежде чем вносить любые изменения в базу данных *Runtime*, рекомендуется сделать резервную копию существующих данных. Желательно также резервировать базу данных Runtime на регулярной основе, чтобы свести к минимуму риск потери данных в случае отказа диска.

При резервировании базы данных все системные таблицы, пользовательские объекты и данные копируются на отдельное устройство, называемое резервным или дамповым. К устройствам дампа относятся дискеты, zip-диски, приводы на магнитной ленте и т.п. Резервирование легко производится при помощи SQL Server Enterprise Manager.

Подробнее см. в главе 9 этого руководства, "Резервирование данных и баз данных".

## Импортирование из InTouch

Функция импорта тэгов позволяет выбрать словарь данных InTouch (TAGNAME.X) и импортировать всю информацию из приложения InTouch в базу данных *Runtime* IndustrialSQL Server.

---

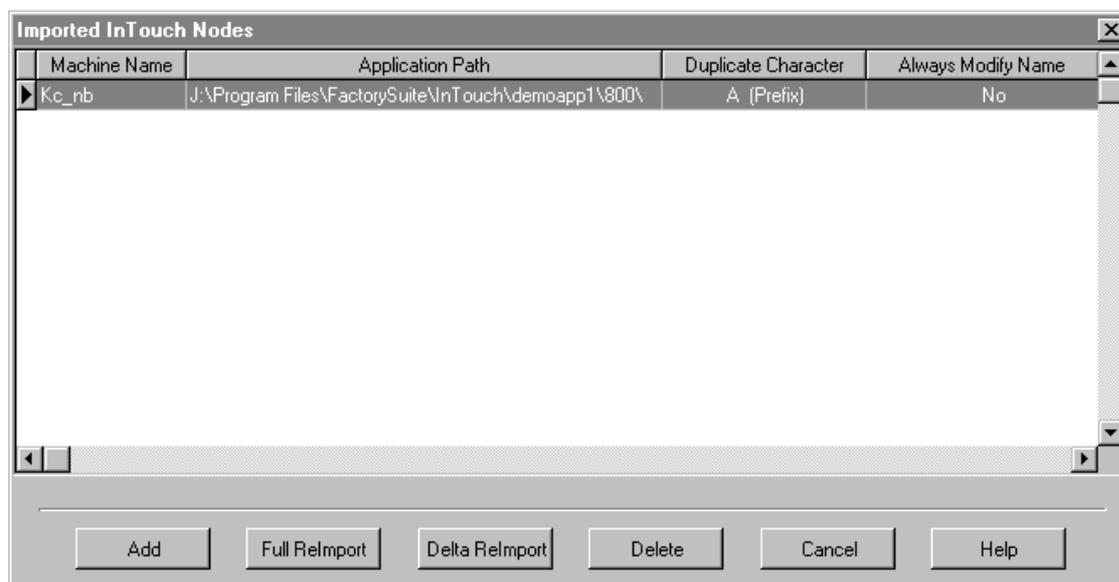
**Примечание.** Импортировать можно базы данных тэгов из InTouch версии 6.0 и выше. Однако дельта-импортирование возможно только для InTouch 7.1. Перед импортированием данных рекомендуется зарезервировать базу данных *Runtime*. Импортировать имеет право только пользователь с уровнем "wwAdmin" или "wwdbo".

---

➤ **Как импортировать данные из файла TAGNAME.X:**

1. Запустите программу InSQL Configure.
2. В меню **Tools** выберите **Import Tags**.

Появится диалоговое окно **Imported InTouch Nodes**. (Если в IndustrialSQL Server в настоящий момент нет импортированных узлов, в этом диалоговом окне вы увидите только информационный текст и кнопку **Add**.)



Значения столбцов **Duplicate Character** и **Always Modify Name** появляются только при многоузловом импортировании. При импортировании одиночного узла в столбце **Duplicate Character** появляется "Нет", а в столбце **Always Modify Name** вообще ничего нет.

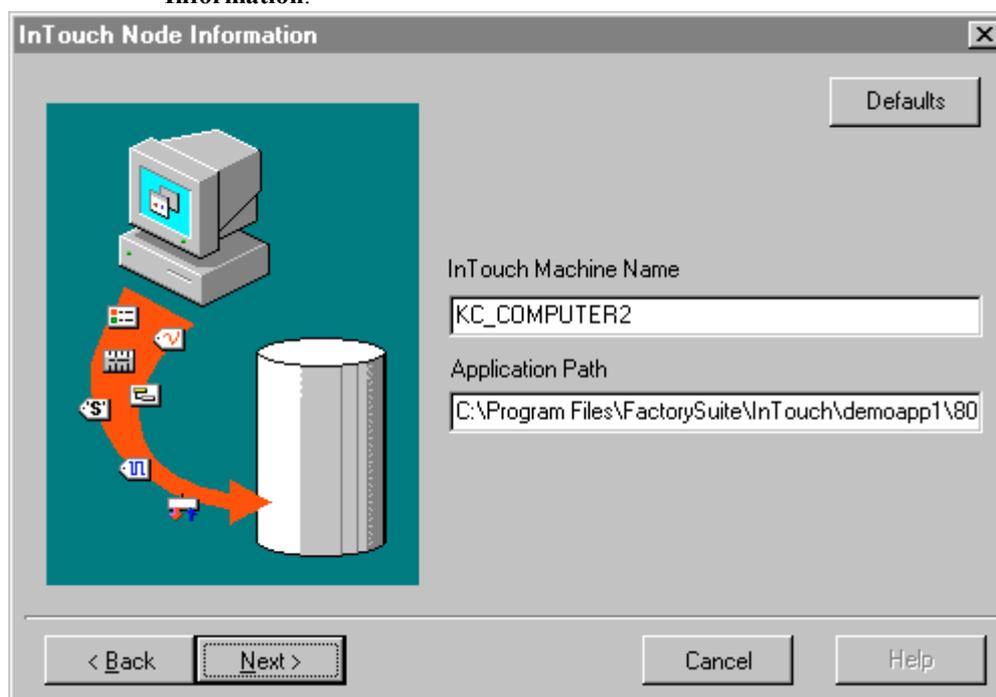
3. Для импортирования узла InTouch и связанных с ним тэгов нажмите кнопку **Add**. В следующем диалоговом окне **Select Tagname.x** найдите файл TAGNAME.X (InTouch версии 6.0 или позже), который нужно импортировать, и нажмите кнопку **Open**. Если импортирование происходит впервые, будет запрошено подтверждение операции.

Для повторного импортирования узла выберите нужный узел и нажмите **Full ReImport**.

Для повторного импортирования только тех тэгов, которые изменились на определенном узле, выберите узел и нажмите **Delta ReImport**.  
Подробнее см. в разделе "Дельта-реимпортирование" выше в этой главе.

Для удаления узла и всех связанных с ним тэгов выберите нужный узел и нажмите **Delete**. Можно удалить только первый импортированный узел, если других импортированных узлов нет; то есть, нельзя удалить первый узел, если импортировано несколько узлов. Появится запрос, требующий подтвердить удаление.

4. После запуска сценария появится диалговое окно **InTouch Node Information**.



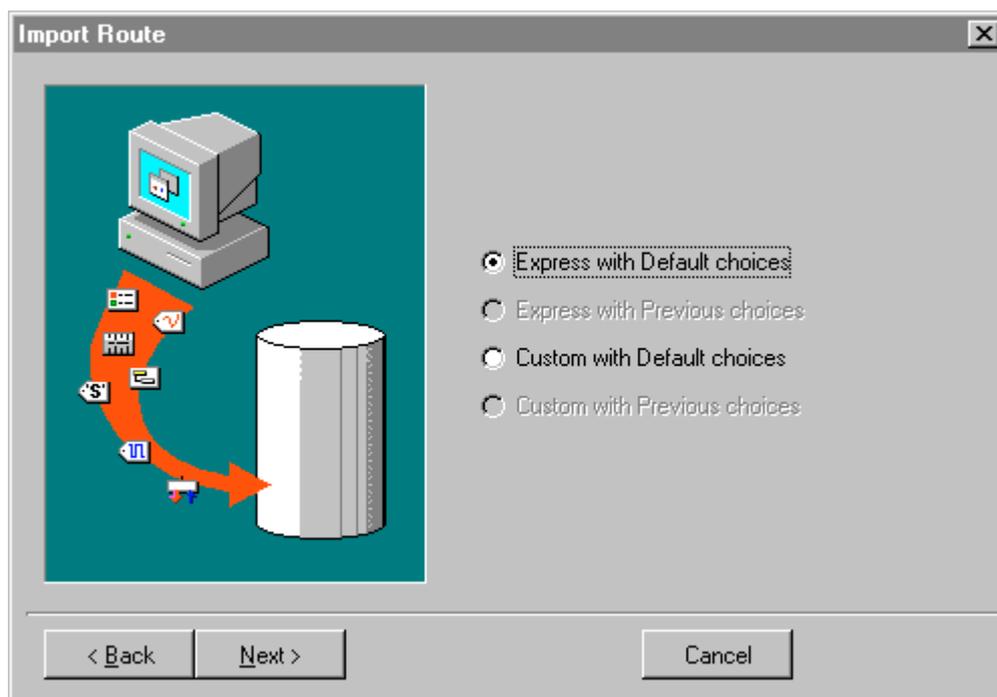
5. Укажите имя компьютера InTouch и путь к приложению InTouch.

(Во время повторного импортирования кнопки в этом диалговом окне недоступны).

При первом импортировании узла InTouch текущие имя машины и путь к приложению InTouch будут отображены по умолчанию. Подробнее см. в разделе "Редактирование имен компьютеров" выше в этой главе.

6. Нажмите **Next**.

Появится диалговое окно **Import Route**:



7. Выберите тип импорта.

Предыдущие настройки (Previous choices) недоступны при исходном импортировании. Если выбран один из стандартных параметров (Default), то при нахождении дублированного тэга к его имени будет добавлен унифицирующий символ.

Чтобы завершить редактирование параметров импорта, можно в любой момент нажать **Finish**. При выборе кнопки **Finish** текущие настройки будут сохранены для остальных настроек.

**Express with Default Choices**

Система выполняет импортирование, используя стандартные параметры. Стандартом являются тэги ввода-вывода со скоростью сохранения 10 секунд. Указывать какую-либо информацию для параметров импортирования не требуется.

**Express with Previous Choices**

Если ранее уже производился импорт InTouch, то новая операция импортирования будет выполняться с использованием стандартных настроек предыдущей операции. Указывать какую-либо информацию для параметров импортирования не требуется.

**Custom with Default Choices**

Позволяет редактировать параметры импорта, основанные на поддерживаемых системой стандартах. Стандартом являются все тэги ввода-вывода со скоростью сохранения 10 секунд. При запуске импортирования поддерживаемые системой стандарты будут отображены. Предлагается либо принять стандартные настройки, либо изменить их.

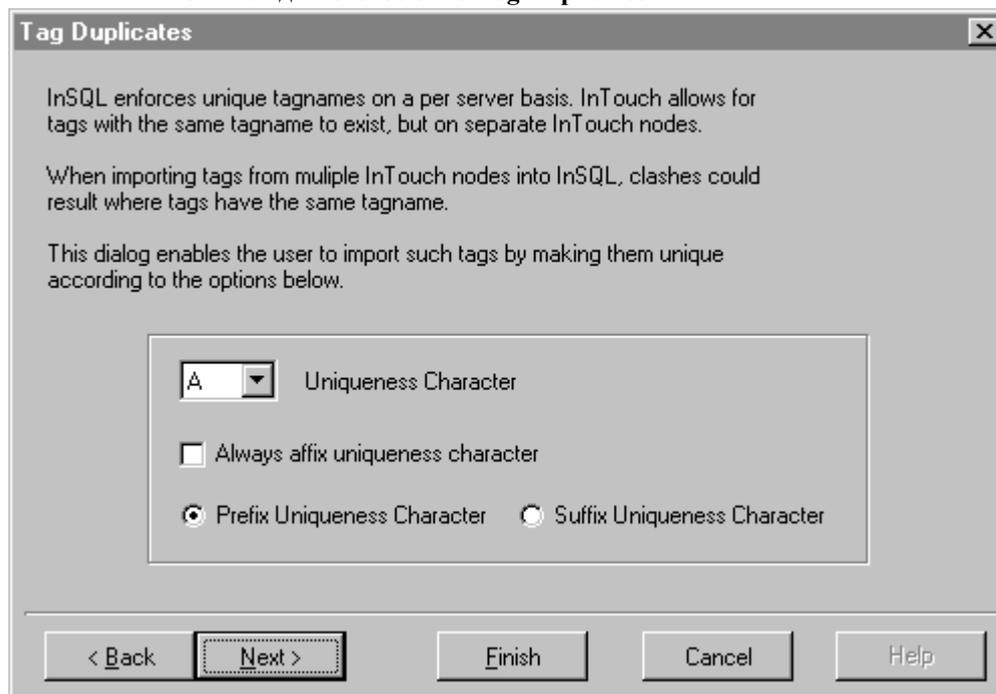
**Custom with Previous Choices**

Позволяет редактировать параметры импорта, основанные на предыдущей операции импортирования.

При запуске импортирования отображается информация предыдущего импортирования. Вы можете принять эти параметры или изменить их.

8. Нажмите **Next**.

Появится диалоговое окно **Tag Duplicates**.



9. Настройте параметры обработки дублируемых имен тэгов. (Во время повторного импортирования параметры в этом диалоговом окне недоступны.)

#### **Uniqueness Character**

Унифицирующий символ от A до Z и от 0 до 9.

#### **Always affix uniqueness character**

Добавляет символ во все имена импортированных тэгов независимо от того, дублируются они или нет. Однако, если при добавлении символа получается дублируемое имя тэга, этот тэг не импортируется. Список всех таких тэгов выводится по завершении импорта.

#### **Prefix uniqueness character**

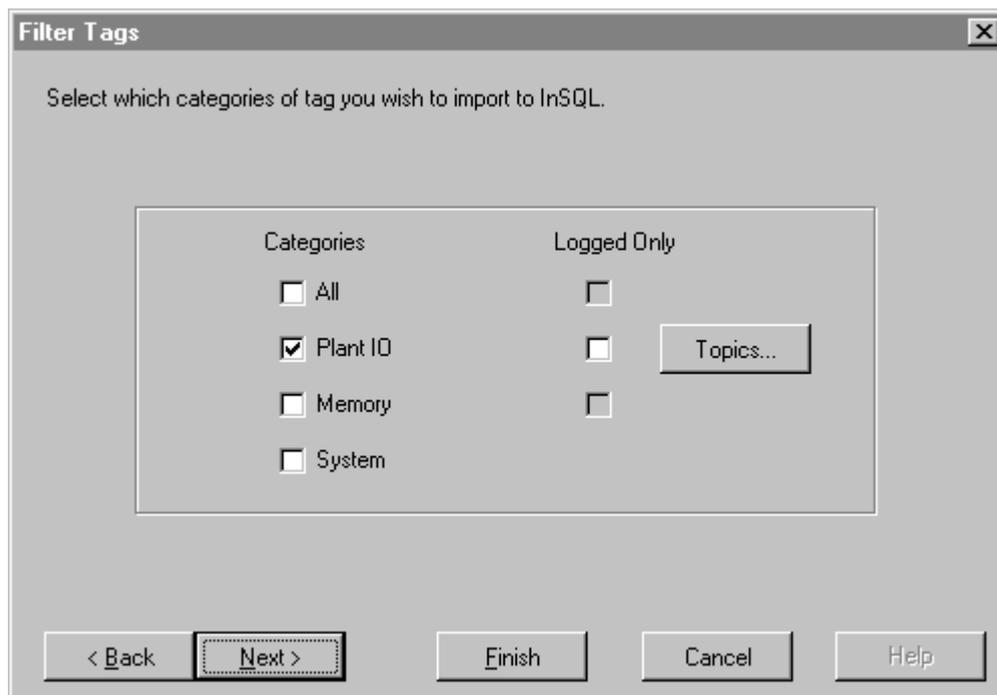
Добавляет символ в начало имени тэга.

#### **Suffix uniqueness character**

Добавляет символ в конец имени тэга.

10. Нажмите **Next**.
11. При выборе любого из **Express**-параметров импортирования появляется диалоговое окно **Final Confirmation**. Нажмите **Next**, чтобы приступить к импортированию.

При выборе любого из **Custom**-параметров появляется диалоговое окно **Filter Tags**.



12. Отметьте категории тэга, который следует импортировать:

**All**

Импортировать всю информацию о тэгах из словаря данных. При выборе этого параметра автоматически включаются все другие категории тэгов.

**Plant IO**

Импортировать только тэги с промышленных серверов ввода-вывода.

**Memory**

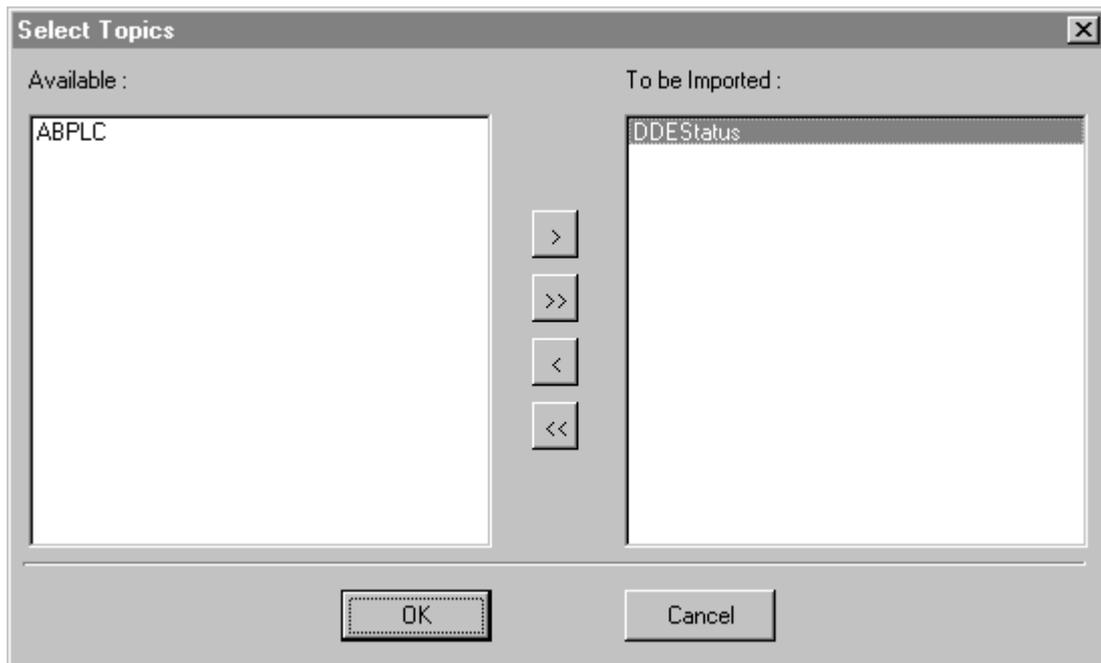
Импортировать только тэги внутреннего типа (внутренний дискретный, внутренний целый, внутренний действительный, внутренний текстовый).

**System**

Импортировать только системные тэги.

13. Для каждой выбранной категории укажите, следует ли включать в нее только те тэги, которые зарегистрированы в InTouch.
14. Если выбрано импортирование не всех тэгов, следует уточнить, какие темы нужно импортировать из категории **Plant IO**. Для этого нажмите кнопку **Topic**.

Появится диалоговое окно **Select Topics**.



- Используя кнопки "вправо" и "влево", переместите нужные темы в окно **To be Imported**. В приведенном примере не будут импортированы тэги, принадлежащие теме ABPLC. Нажмите **OK**.

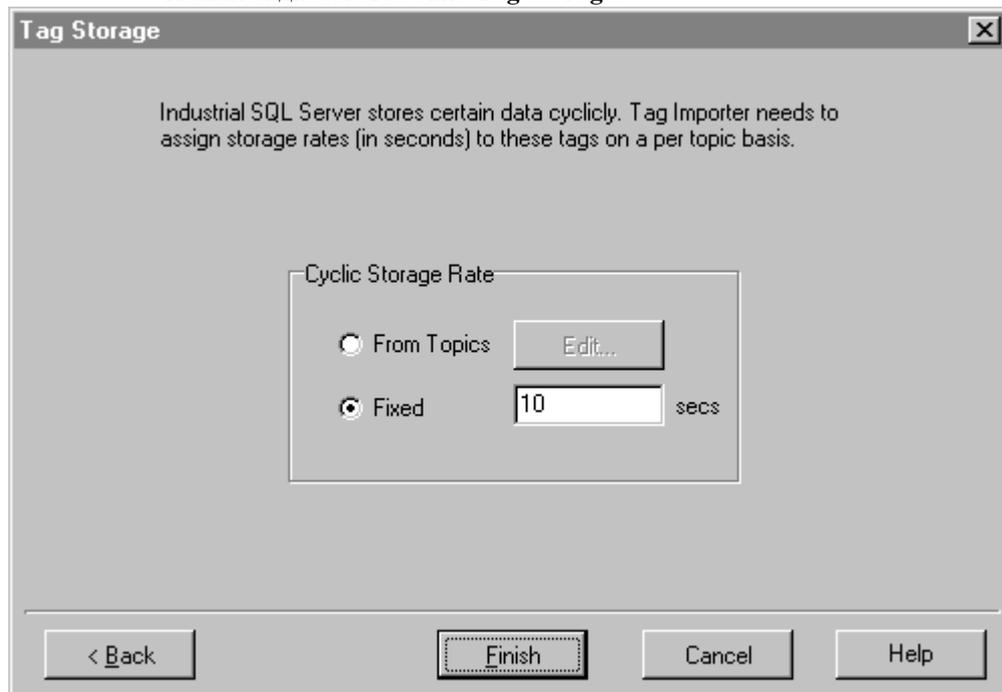
---

**Примечание.** При повторном импортировании в IndustrialSQL Server будут обновлены только тэги выбранных тем. Темы из предыдущего импортирования не будут изменены.

---

- Нажмите **Next**.

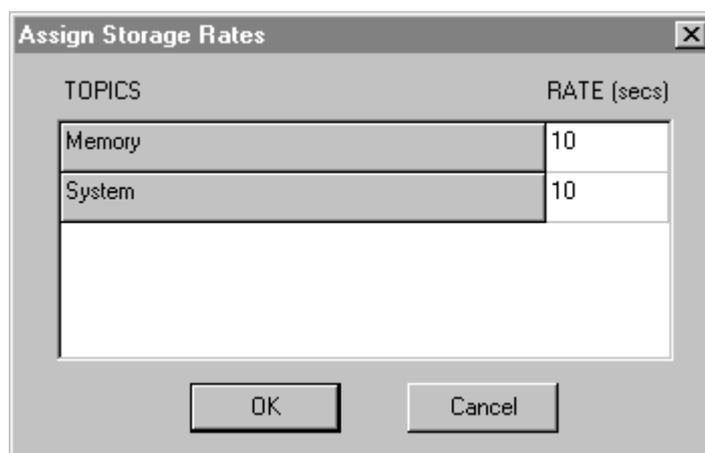
Появится диалоговое окно **Tag Storage**.



17. Настройте скорости циклического сохранения. Скорость циклического сохранения есть временной интервал между сохраненными значениями тэгов, не зависящий от того, изменились значения или нет. Например, если скорость сохранения установлена на 10, значения всех тэгов будут сохраняться в IndustrialSQL Server каждые 10 секунд.

Для использования глобальной фиксированной скорости сохранения (превышающей изначально установленную для этой темы), нажмите **Fixed** и введите скорость в поле секунд.

Для использования скорости сохранения импортированных тем нажмите **From Topics**. Для редактирования любой темы нажмите **Edit**. Появится диалоговое окно **Assign Storage Rate**, в котором можно изменить значение скорости:



18. Нажмите **Finish**, чтобы закончить настройку импортирования.
19. Появляется диалоговое окно последнего подтверждения, после чего начнется процесс импорта.

Когда процесс импорта тем завершится, появится диалоговое окно **Information**.



20. Нажмите **OK**.
21. Переходите к следующему разделу, "[Повторная инициализация системы](#)", где содержится информация по перезапуску системы хранения.

## Повторная инициализация системы

После импортирования следует повторно инициализировать систему. Изменения в базе данных *Runtime* IndustrialSQL Server не имеют эффекта, пока не выполнена повторная инициализация. Повторная инициализация вызывает выключение системы хранения IndustrialSQL Server и самого сервера, а затем снова запускает систему хранения, автоматически включая сервер.

Для повторной инициализации можно использовать программу InSQL Configure.

Подробнее см. в главе 2, "InSQL Configure", в этом руководстве администратора.

## Повторное импортирование

Для повторного импортирования словаря тэгов InTouch не требуется никаких особых настроек. Повторное импортирование можно выполнять только с тем же узлом InTouch. Следуйте инструкциям по импорту TAGNAME.X.

Если при процессе повторного импортирования выбрать импорт не всех тем, то будут обновлены только тэги выбранных тем IndustrialSQL Server. Остальные темы останутся без изменения.

Можно также повторно импортировать те тэги, которые изменились для конкретного узла InTouch со времени последнего импортирования.

Инструкции по импортированию файла TAGNAME.X см. в разделе "Импортирование из InTouch" выше в этой главе.

## IndustrialSQL как поставщик архива для InTouch

Данные, хранящиеся в базе *Runtime* IndustrialSQL Server, можно выбирать и просматривать из любого приложения InTouch, настроенного для использования IndustrialSQL в качестве поставщика архива. Поставщиком архива считается любая внешняя архивная база данных, к которой имеет доступ InTouch.

Подробнее об поставщиках архива InTouch см. в *Руководстве пользователя InTouch*.

## Элементы ActiveX системы IndustrialSQL Server

Программа InTouch WindowViewer поддерживает объекты ActiveX. При создании графического пользовательского интерфейса в программе WindowMaker можно выбрать любые элементы ActiveX из IndustrialSQL Server. В среде выполнения (WindowViewer) элементы ActiveX получают значения тэгов и другие данные из системы IndustrialSQL Server.

Подробнее см. в *Руководстве по элементам ActiveX IndustrialSQL Server*.  
Подробнее об использовании элементов ActiveX в InTouch см. в *Руководстве пользователя InTouch*.

## Использование InTouch SQL Access Manager

Программа InTouch SQL Access Manager позволяет:

- Перемещать данные из базы SQL в приложение InTouch.
- Перемещать данные реального времени, состояния алармов или архивные данные из InTouch в базу SQL.

SQL Access Manager используется для создания и связывания столбцов базы данных с именами тэгов в базе данных тэгов InTouch. Такое связывание позволяет SQL Access Manager напрямую управлять данными в базе данных.

Функции SQL Access Manager могут использоваться любым сценарием InTouch. Эти функции позволяют создавать новые таблицы, вставлять в таблицы новые записи, редактировать существующие записи таблицы, очищать и удалять таблицы, выбирать записи, перемещаться по таблице и т.п. Выполнение этих функций зависит от ввода оператора, изменения значения тэга или от наличия определенных условий.

Подробнее см. в *Руководстве пользователя InTouch SQL Access Manager*.

SQL Access соединяется с IndustrialSQL Server с помощью драйвера ODBC Microsoft SQL Server. Перед запуском SQL Access следует установить соединение через ODBC с IndustrialSQL Server.

---

**Примечание.** Если приложение InTouch имеет доступ к данным IndustrialSQL Server 7.0, следует изменить строку соединения ODBC для выполнения запросов к базе данных IndustrialSQL Server 7.1. Например, для протокола "named pipes" строку соединения следует изменить с `\\<ИмяСервера>\pipe\sql\query` на `\\<ИмяСервера>\pipe\wwsql`. Однако изменять строку соединения необязательно, если все запросы имеют синтаксис OLE DB.

---

Подробнее об использовании ODBC см. в *Руководстве администратора системы FactorySuite*.

## Создание соединения с базой данных

Для того, чтобы приложение InTouch имело доступ к базе данных *Runtime* IndustrialSQL Server, в среде выполнения (WindowViewer) должно быть установлено соединение. Соединение с IndustrialSQL Server может быть установлено путем запуска сценария InTouch, содержащего функцию **SQLConnect()**. При использовании этой функции следует указать:

- Имя источника данных (IndustrialSQL Server).
- Базу данных (*Runtime*).
- Имя пользователя и пароль (учетную запись пользователя IndustrialSQL Server).

Например, можно создать сценарий, автоматически устанавливающий соединение с IndustrialSQL Server при запуске приложения InTouch.

Синтаксис может выглядеть так:

```
ResultCode = SQLConnect( ConnectionId,  
"DSN=InSQL2;DB=Runtime;UID=wwUser;PWD=wwUser" );
```

Когда соединение с IndustrialSQL Server успешно установлено, выдается целое значение параметра **SQLConnect()** *ConnectId*. Значение *ConnectId* может быть использовано во всех сценариях IndustrialSQL Server.

Для отсоединения от базы данных используется функция **SQLDisconnect()**. Например, следующий сценарий (подразумевающий использование кнопок) может быть использован для отсоединения от сервера:

```
ResultCode = SQLDisconnect( ConnectionId,  
"DSN=InSQL2;DB=Runtime;UID=wwUser;PWD=wwUser" );
```

Заметьте, что указывать источник данных здесь необязательно; сценарий использует значение *ConnectId*, полученное при выполнении функции **SQLConnect()**.

Подробнее см. в *Руководстве пользователя InTouch SQL Access Manager*.

## Использование сложных запросов

Использование функций SQL Access в сценариях InTouch позволяет делать более точными сложные запросы и выражения SQL для выборки информации из базы данных IndustrialSQL. Например, для получения данных из таблицы *Annotation* может использоваться следующий сценарий:

```
ResultCode = SQLSetStatement( ConnectionId, "SELECT ИмяТэга,  
Content FROM Тэг, Annotation" );
```

```
ResultCode = SQLAppendStatement( ConnectionId, "WHERE  
Annotation.Value = 1500 AND Tag.TagName = Annotation.TagName"  
);
```

SQL Access для InTouch поддерживает также чтение сценариев SQL из файлов, созданных при помощи сторонних инструментов, таких как Microsoft Access.

Подробнее см. в *Руководстве пользователя InTouch SQL Access Manager*.

# Управление системой хранения

В этой главе описана система хранения IndustrialSQL Server и содержится информация о типах хранения данных, архивных блоках и их нотации, а также об областях хранения.

## Содержание

- О системе хранения
- Хранение данных конфигурации
- Хранение архивных данных
- Требования к размеру диска
- Запуск системы хранения
- Операции системы хранения
- Резервирование архивных данных
- Поддержка перевода часов на летнее/зимнее время
- Настройка дельта-хранения для быстро изменяющихся тэгов

## О системе хранения

В системе хранения IndustrialSQL Server сохраняются значения аналоговых, дискретных, текстовых, сложных тэгов и тэгов событий. Эта система поддерживает два метода хранения данных: циклический и дельта. Поддерживаемые парадигмы хранения значений тэгов приведены в следующей таблице:

Тип тэга	Циклическое	Дельта	Таблица
Аналоговый	ДА	ДА	<i>AnalogHistory</i>
Дискретный	НЕТ	ДА	<i>DiscreteHistory</i>
Текстовый	НЕТ	ДА	<i>StringHistory*</i>
Сложный	НЕТ	НЕТ	<i>ComplexHistory*</i>
Событие	НЕТ	НЕТ	<i>EventHistory*</i>

\* *ComplexHistory*, *StringHistory* и *EventHistory* являются нормальными таблицами SQL Server.

### Циклическое хранение

Циклическим называется метод хранения данных, основанный на временном интервале. Например, можно сохранять значение аналогового тэга каждую секунду. Циклический метод используется для сохранения аналоговых значений, когда эти данные быстро изменяются.

У каждого тэга есть своя скорость сохранения. Для любого тэга можно указать скорость сохранения от одной секунды и более. Скорость, с которой будет сохраняться значение тэга, зависит от параметра разрешения, определяющего временной интервал сохраняемых данных. Сохранение данных с очень маленьким интервалом вызовет сбор очень большого объема данных за очень малое время. Сохранение данных со слишком большим интервалом, в то же время, может привести к тому, что важные данные будут пропущены. Например, если для тэга "BOILER1TEMP" установлена скорость сохранения в 1 неделю, значение этого тэга будет сохраняться лишь раз в неделю.

### Дельта-хранение

Дельта-хранение есть метод сохранения данных, основанный на изменении значения. Дельта-метод позволяет сохранять данные только при изменении значения тэга. Дельта-хранение также называется "хранение методом исключения". Дельта-хранение используется для сохранения дискретных или аналоговых значений, которые остаются константами на долгие периоды времени. Например, вы не хотите сохранять значение дискретного тэга "PUMPON" каждые десять секунд, если PUMP находится в положении ON на протяжении нескольких месяцев. Значение сохраняется вместе с меткой даты/времени изменения с точностью до 1 мс.

---

**Примечание.** Microsoft SQL Server сохраняет значения с разрешением в 3,33 мс. Для данных, сохраняемых с разрешением в 1 мс, метка даты/времени берется из Microsoft SQL Server. При обработке запроса могут быть получены три строки с одинаковыми метками времени, но, возможно, с разными значениями. Если данные выданы в беспорядке (строки смешались), то невозможно определить, в каком порядке они должны были быть.

---

## Активный образ

Активный образ IndustrialSQL Server — это ячейка памяти, в которой временно содержатся значения полученных данных.

Скорость поступления данных в активный образ зависит от скорости изменения получаемых значений определенного тэга. (Эта скорость получения значений активным объектом НЕ относится к значениям, хранящимся в столбце *AcquisitionRate* таблицы *Tag*). Например, если имеется тэг "Tag1" со скоростью сбора равной 3 секундам, система IndustrialSQL Server будет считывать значения этого тэга из активного объекта каждые 3 секунды. Эти значения будут поступать в активный образ до тех пор, пока их количество не достигнет предела числа значений. Первоначально этот предел установлен на 65 значений (выборок).

Количество выборок тэга сохраняется в столбце *SamplesInActiveImage* таблицы *Tag*. Значение NULL или любое другое значение менее 65 для этого столбца показывает, что активный образ использует стандарт в 65 выборок.

Когда достигнут предел активного образа, старейшие значения будут перезаписываться новыми. Перезапись устаревших тэгов освобождает место для новых значений всех тэгов со стандартным количеством в 65 значений, которые могут сохраняться в активном образе в любое заданное время. Все сохраненные в активном образе значения могут не быть записаны на диск — это зависит от скорости сохранения (из столбца *скоростей Сохранения* в таблице *Тэгов*). Если выборки поступают в активный образ со скоростью большей, чем скорость сохранения, значения тэгов будут приниматься системой с большим разрешением, чем то, с каким они будут сохраняться.

Для предотвращения перезаписи значений данных до сохранения их на диск, количество выборок в активном образе должно быть увеличено, чтобы убрать временной провал между получением значений и их сохранением на диск.

## Автоматическое изменение размера активного образа

Система периодически пересчитывает оптимальное количество выборок для каждого тэга в зависимости от скорости сбора данных. Начиная с одной минуты после запуска и далее через каждые пять минут, система тестирует объем памяти, выделенный под активный образ, на предмет необходимости его увеличения. Система подсчитывает среднее время для выборки тэга, учитывая метки времени первой и последней выборки в активном объекте и количество выборок. При необходимости система увеличит количество выборок для проведения 65-секундного кэширования. Если же подсчитанное время больше 65 секунд, система не будет изменять стандартные 65 значений (NULL в столбце *SamplesInActiveImage*).

Короче говоря, при получении более 65 изменений значения за 65-секундный интервал, система IndustrialSQL Server увеличит количество *SamplesInActiveImage* в таблице *Tag* и увеличит объем памяти для хранения дополнительных выборок быстро изменяющихся значений тэгов.

Количество выборок в активном образе никогда не уменьшается автоматически, но его можно уменьшить вручную при помощи хранимой процедуры **ww\_SetAISamples**.

Подробнее о хранимой процедуре **ww\_SetAISamples** см. в *Справочном руководстве IndustrialSQL Server*.

Активный образ можно использовать для оптимизации системы, т.к. система сбора данных проверяет активный образ на наличие значений с требуемыми метками времени, удовлетворяющими запросу, вместо отправки их прямо в архивные блоки.

Например, требуется создавать ежечасные отчеты за неделю производства, основанные на тэге 'SysTimeHour', а затем может потребоваться увеличить количество выборок в активном образе для этого тэга на 144 ( $24 * 7 = 144$ ). Таким образом, система не обращается к диску для получения данных тэга 'SysTimeHour', и общее время получения данных и загрузка CPU уменьшаются.

Чтобы выключить автоматическое увеличение активного образа, следует установить значение реестра **AIAutoResize** на 0. Также можно изменить скорость, с которой система пересчитывает выборки для каждого тэга в активном образе, изменив значение реестра **AIResizeSecInterval**.

Подробнее см. в приложении А.

## Влияние активного образа на выборку данных

При обработке запроса система IndustrialSQL Server всегда сначала ищет данные в активном образе. При выборке данных можно заметить различие между значениями тэгов, полученными за один и тот же период, в зависимости от того, когда был подан запрос. Это относится к тэгам, сохраняемым циклически или методом исключения (дельта-методом) со значением временной мертвой зоны.

Например, если запрос подан между 12/6/98 1:00:00 и 12/6/98 1:02:00, а данные в этот период находились в активном объекте, они будут выданы с тем разрешением, с которым были получены.

Но если тот же запрос будет подан несколько дней спустя, когда значения будут храниться уже в архивных блоках на диске, то столь же высокого разрешения может не быть. Когда данные перемещаются из активного объекта на диск с учетом скорости сохранения, некоторые значения могут быть утеряны, если скорость сохранения меньше скорости сбора. Может показаться, что эти значения "утрачены", тогда как они просто никогда не были настроены для сохранения в первую очередь.

## Хранение данных конфигурации

Данные конфигурации системы IndustrialSQL Server — это данные, которые определяют работу всей системы. Данные конфигурации статичны; они не изменяются в результате производственных операций. Данные конфигурации IndustrialSQL Server содержат следующую информацию:

- Количество узлов хранения.
- Определения сервера ввода-вывода.
- Определения тэгов
- Определения пользователей базы данных и групп пользователей.
- Настройка именных пространств (иерархия объектов базы данных).

Данные конфигурации хранятся в различных таблицах базы данных *Runtime* IndustrialSQL Server. База данных *Runtime* хранится как файл базы данных Microsoft SQL Server.

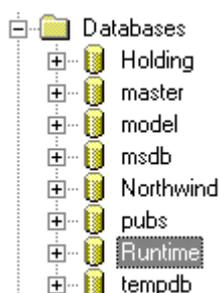
Подробнее см. в Справочном руководстве IndustrialSQL Server.

## Просмотр файлов базы данных

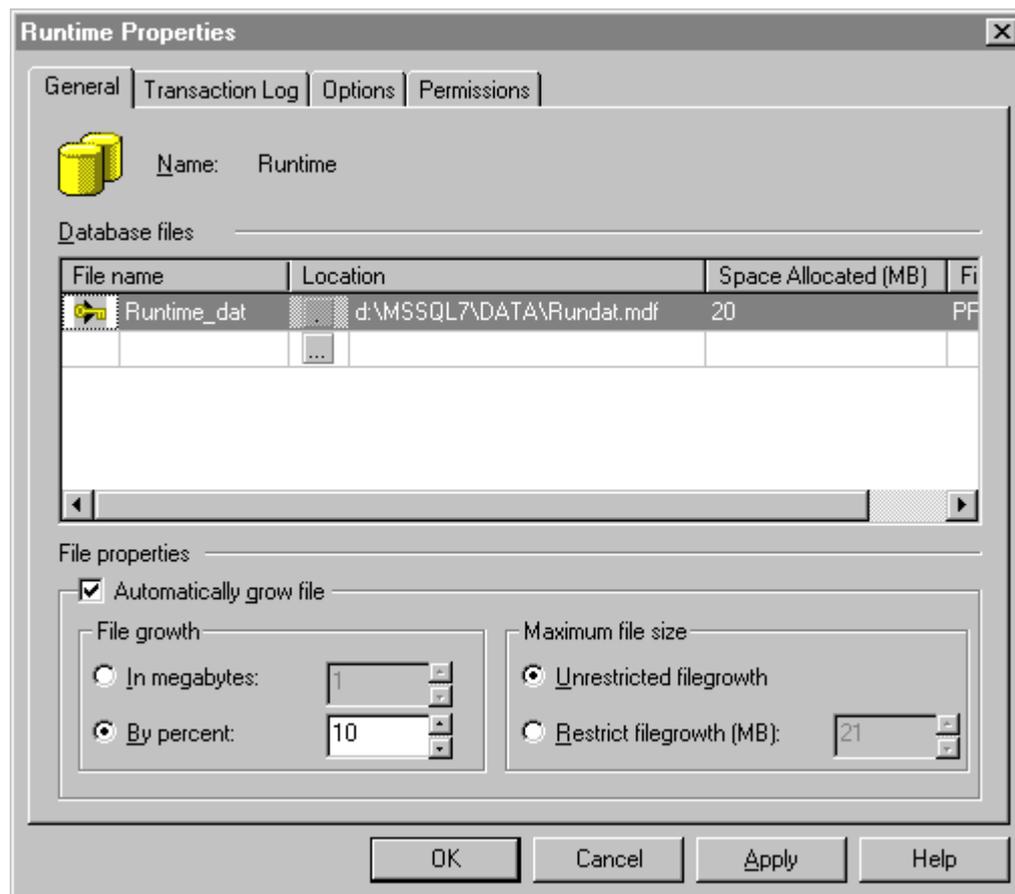
Файл базы данных — это файл, в котором Microsoft SQL Server хранит данные. При установке IndustrialSQL Server создаются два файла баз данных: RUNDAT.MDF и RUNLOG.MDF. Базы данных *Runtime* и *Holding* содержатся в RUNDAT.MDF, а связанные с ними файлы журналов содержатся в RUNLOG.MDF.

### ➤ Как просмотреть детали файла базы данных:

1. Запустите Microsoft SQL Server Enterprise Manager.
2. Разверните IndustrialSQL Server.
3. Разверните папку **Databases**.



4. Нажмите правой кнопкой мыши на базе данных *Runtime*. При появлении меню выберите **Properties**.
5. Появится диалоговое окно **Runtime Properties**.



Это окно показывает, где хранится файл базы данных (D:\MSSQL7\DATA\RUNDAT.MDF). Заметьте, что текущий размер устройства = 20 Мб.

**Совет.** Для просмотра файла базы данных в Проводнике Windows загляните в каталог \DATA, расположенный обычно в каталоге Microsoft SQL Server: C:\MSSQL7\DATA.

6. Для просмотра тех же деталей журнала транзакций откройте вкладку **Transaction Log**.
7. Использование параметров в окне **Runtime Properties** дает возможность разворачивать и сворачивать базу данных и пересчитывать свободное пространство базы данных для журнала транзакций. Также можно устанавливать параметры базы данных для того, чтобы предоставить или отменить доступ для пользователей и групп.

Подробнее см. в документации Microsoft.

8. Нажмите кнопку **Cancel**.

## Хранение архивных данных

Архивные данные, в отличие от данных конфигурации, постоянно сохраняются во время производственных операций. Данные "динамичны" в том смысле, что постоянно изменяются. Производственные данные, получаемые серверами ввода-вывода, сохраняются в таблицах расширения (при использовании OLE DB это — удаленные таблицы) IndustrialSQL Server. К этим таблицам относятся: *AnalogHistory*, *AnalogLive*, *AnalogWideHistory*, *DiscreteHistory*, *DiscreteLive*, *DiscreteWideHistory*, *HistoryBlock* и *StringLive*.

Таблицы расширения (удаленные таблицы) являются обычными таблицами SQL Server. Данные обычных таблиц SQL Server хранятся в файлах базы данных (.MDF), управляемых сервером SQL. Данные, представляемые через удаленные таблицы *AnalogLive*, *DiscreteLive* и *StringLive*, получают из активного образа IndustrialSQL Server. Данные в удаленных таблицах (расширения) *AnalogHistory* и *DiscreteHistory* сохраняются в блоках данных, называемых архивными блоками, на диске компьютера, на котором расположен IndustrialSQL Server, и управляются они системой хранения IndustrialSQL Server. Когда данные получают с производственного участка, размер этих таблиц возрастает, причем этот рост ограничен только размером жесткого диска, на котором расположена база данных. Таблицы *AnalogWideHistory* и *DiscreteWideHistory* являются модификациями таблиц *AnalogHistory* и *DiscreteHistory*.

При запуске системы создается один архивный блок. Затем новые архивные блоки автоматически создаются на каждом "интервале блока". Интервал блока определяется значением параметра *SecondsPerBlock* следующего ключа реестра:

```
HKEY_LOCAL_MACHINE
  \SOFTWARE
    \WONDERWARE
      \InSQL
        \Storage
```

Значение параметра *SecondsPerBlock* должно быть кратно 1 часу. По умолчанию оно равно 24 часам. В каждом архивном блоке содержатся сжатые и индексированные файлы, которые составляют архив собранных производственных данных.

Не рекомендуется использовать блоки длиной менее 3 часов.

[Подробнее см. ниже в разделе "Поддержка перевода часов на летнее/зимнее время".](#)

---

**Примечание.** Система поддерживает 26 блоков в день. Если ограничение превышено, система начнет перезаписывать данные в первом блоке дня. InSQL Control выдаст предупреждающее сообщение, когда ограничение будет почти достигнуто. Для уменьшения числа архивных блоков, созданных за день, увеличьте стандартный размер архивного блока (если он меньше 24 часов).

---

## Расположение архивного блока

Место хранения архивных блоков определяется информацией в таблице *StorageLocation* в базе данных *Runtime*. Эта информация также хранится как значения в следующем ключе реестра:

```
HKEY_LOCAL_MACHINE
  \SOFTWARE
    \WONDERWARE
      \InSQL
        \Datapaths
```

Значения ключа *Datapaths* описаны в следующей таблице:

Ключ	Описание
DynData	Циклическая область хранения. Используется для хранения архивных блоков.
AltData	Альтернативная область хранения. Используется как вторичная циклическая область хранения.
BufferDisk	Буферная область хранения. Содержит данные, полученные с носителей резервных копий (таких как DAT-кассеты).
WormDisk	Постоянная область хранения. Используется для постоянного хранения копий архивных блоков.

Подробнее см. ниже в разделе "Области хранения архивных блоков".

---

**Примечание.** Значения в ключе *Datapaths* автоматически получаются службой *InSQL Control* из информации, полученной с помощью программы *InSQL Configure*. **Не изменяйте эти значения в реестре.**

---

При разработке стратегии хранения данных *IndustrialSQL Server* удостоверьтесь, что на диске достаточно свободного места для хранения производственных данных за требуемый отрезок времени. Во время установки потребуется указать путь для циклической, буферной и постоянной областей хранения (альтернативная область хранения может быть определена при помощи *InSQL Configure*).

## Изменение мест хранения

Для просмотра и редактирования мест хранения можно использовать программу *InSQL*. С ее помощью можно настроить:

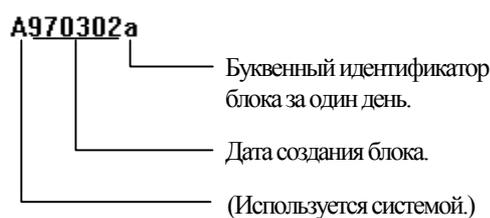
- Пути к местам хранения.
- Порог, при достижении которого система начнет освобождать дисковое пространство для новых архивных блоков.
- Ограничение объема данных, сохраняемых в указанном месте.

Подробнее см. в главе 2, "*InSQL Configure*".

## Нотация архивного блока

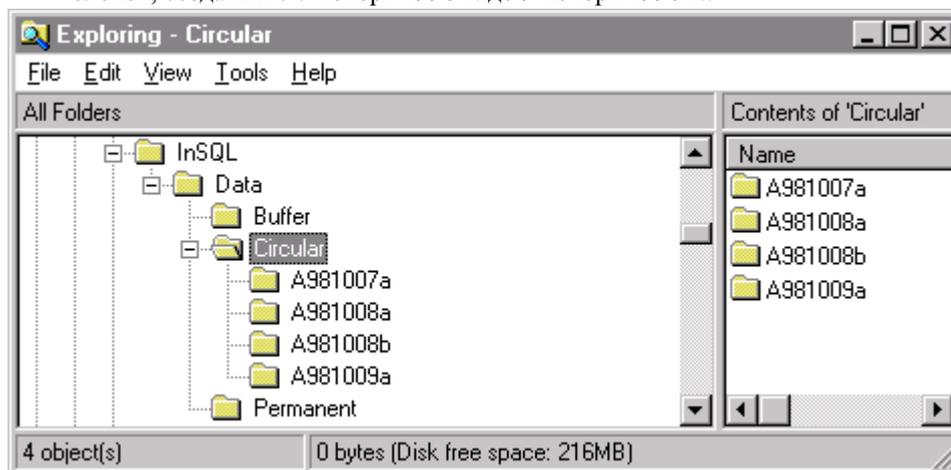
IndustrialSQL Server сохраняет собранные данные на диск в подкаталоги, называемые блоками. Архивные блоки создаются, когда система запускается, после окончания заранее определенного и указанного в реестре интервала времени

Каждый блок является отдельным каталогом, идентифицируемым меткой даты/времени и буквенным суффиксом. Буквенный суффикс используется для различения блоков созданных в один день в результате, например, многочисленных перезапусков системы или в случае, когда интервал создания блоков установлен на время меньше одного дня.



**Примечание.** Система поддерживает 26 архивных блоков в день. Если этот предел достигнут, система начнет перезаписывать данные в первом архивном блоке за текущий день. InSQL Control выдаст предупреждающее сообщение, когда ограничение будет почти достигнуто. Для уменьшения числа архивных блоков, созданных за день, увеличьте стандартный размер архивного блока (если он меньше 24 часов).

Далее показан типичный набор архивных блоков, содержащих данные для блоков, созданных с 7 ноября 1998 г. до 9 ноября 1998 г.:



Блоки имеют размер 24 часа. Система хранения была перезапущена 8 ноября 1998 г., что вызвало появление двух блоков за этот день. Эти два блока в сумме дают 24 часа.

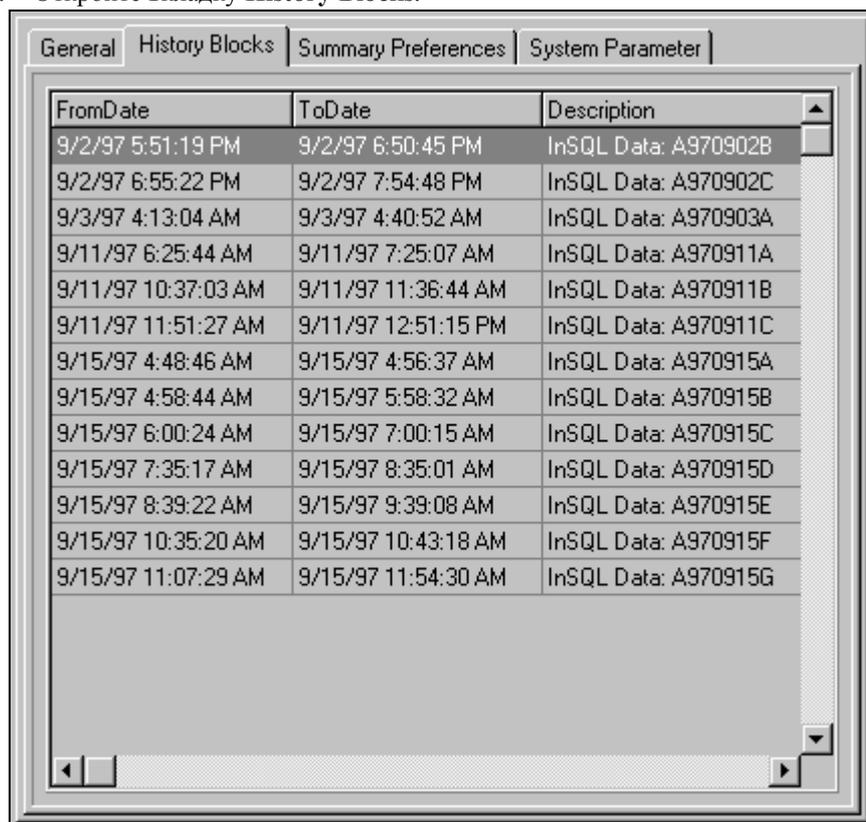
**Примечание.** Microsoft SQL Server кэширует ранее полученные данные для использования в последующих запросах во время текущего сеанса. Если архивные блоки были каким-то образом увеличены (например, при переводе часов на зимнее/летнее время или при копировании архивных блоков), клиент должен разъединиться и вновь соединиться с сервером для обновления и получения новейших данных.

## Просмотр архивных блоков в InSQL Configure

Для просмотра всех архивных блоков можно использовать программу InSQL Configure.

➤ **Как просмотреть архивные блоки:**

1. Запустите InSQL Configure.
2. В браузере выберите сервер, на котором вы хотите просмотреть архивные блоки.
3. Откройте вкладку **History Blocks**.



FromDate	ToDate	Description
9/2/97 5:51:19 PM	9/2/97 6:50:45 PM	InSQL Data: A970902B
9/2/97 6:55:22 PM	9/2/97 7:54:48 PM	InSQL Data: A970902C
9/3/97 4:13:04 AM	9/3/97 4:40:52 AM	InSQL Data: A970903A
9/11/97 6:25:44 AM	9/11/97 7:25:07 AM	InSQL Data: A970911A
9/11/97 10:37:03 AM	9/11/97 11:36:44 AM	InSQL Data: A970911B
9/11/97 11:51:27 AM	9/11/97 12:51:15 PM	InSQL Data: A970911C
9/15/97 4:48:46 AM	9/15/97 4:56:37 AM	InSQL Data: A970915A
9/15/97 4:58:44 AM	9/15/97 5:58:32 AM	InSQL Data: A970915B
9/15/97 6:00:24 AM	9/15/97 7:00:15 AM	InSQL Data: A970915C
9/15/97 7:35:17 AM	9/15/97 8:35:01 AM	InSQL Data: A970915D
9/15/97 8:39:22 AM	9/15/97 9:39:08 AM	InSQL Data: A970915E
9/15/97 10:35:20 AM	9/15/97 10:43:18 AM	InSQL Data: A970915F
9/15/97 11:07:29 AM	9/15/97 11:54:30 AM	InSQL Data: A970915G

4. В окне появятся детали каждого архивного блока.  
[Подробнее см. в главе 2, "InSQL Configure"](#).

## Области хранения архивных блоков

Существует четыре типа областей хранения архивных блоков: циклическая, альтернативная, буферная и постоянная. Путь к этим областям хранения (кроме альтернативной) изначально определен во время установки и хранится в базе данных и в реестре.

[Подробнее см. в главе 2, "InSQL Configure."](#)

---

**Примечание.** Значения в ключе Datapaths автоматически получают службой InSQL Control из информации, полученной с помощью программы InSQL Configure. **Не изменяйте эти значения в реестре.**

---

### Циклическая область хранения

Циклическая область является основной областью хранения данных. При первом запуске системы хранения IndustrialSQL Server создается первый архивный блок и в него начинают записываться данные. Блок архивных производственных данных сохраняется как подкаталог в каталоге циклической области хранения.

Циклическая область хранения состоит из единой области, запись в которую производится методом "циклического буфера". Когда свободное дисковое пространство на диске, содержащем циклическую область хранения, превышает минимальный порог, самые старые данные удаляются из этой области хранения и заменяются новыми. Порог для циклической области хранения устанавливается в столбце *MinMBThreshold* таблицы *StorageLocation* и также в реестре.

Размер циклической области хранения на диске также может быть ограничен. Когда размер содержимого циклической области хранения достигает или превышает эту границу, старейшие данные будут удалены. Предел размера для циклической области хранения хранится в столбце *MaxMBSize* таблицы *StorageLocation*, а также в реестре.

При удалении данных из циклической области хранения эти данные могут быть перемещены в альтернативную область хранения, если таковая определена. Пока дисковое пространство циклической области хранения занято до предела, самые старые данные будут постоянно переноситься в альтернативную область хранения для освобождения места под новые архивные данные.

В обязанности системного администратора входит периодическое отслеживание свободного дискового пространства и резервирование архивных блоков на носителях, таких как DAT-кассеты.

## Альтернативные области хранения

Когда свободное пространство в циклической области хранения превышает заданный порог, или размер каталога циклической области превышает указанный максимальный размер, система IndustrialSQL Server начнет перемещать старейшие архивные блоки в эту область вместо их автоматического удаления. Когда данные удаляются из циклической области хранения для освобождения места под новые архивные данные, они перемещаются в определенные альтернативные области. Альтернативные области хранения пронумерованы. Блок данных последовательно перемещается через альтернативные области, пока в конце концов он не попадает в последнюю область хранения, после чего эти данные все же удаляются из системы.

---

**Примечание.** В данный момент поддерживается только одна альтернативная область хранения.

---

Как минимум, альтернативная область хранения должна располагаться на логическом устройстве отличном, чем циклическая область хранения. (Желательно использовать другой раздел диска, а еще лучше полностью отдельную систему). Эта область хранения необязательна.

Порог альтернативной области хранения указан в реестре.

## Постоянные области хранения

Постоянная область хранения используется для хранения важных данных, которые нельзя перезаписывать. В постоянные области хранения обычно располагаются на устройствах однократной записи (WORM). Система IndustrialSQL Server не будет делать попытки удаления данных из этой области. Данные в постоянной области хранения просматриваются и доступны вместе с данными в циклической области хранения.

Для перемещения архивных блоков в эту область хранения используется расширенная хранимая процедура **xp\_DiskCopy**.

Подробнее о процедуре **xp\_DiskCopy** см. в главе 9, "Резервирование данных и баз данных".

## Буферные области хранения

Буферные области хранения используются для временных целей, таких как выборка из сжатых данных. Эта область хранения может находиться на том же диске, где расположена циклическая область, или на другом. Данные буферной области можно просматривать вместе с данными циклической области хранения. IndustrialSQL Server никогда не удаляет данные из этой области.

## Требования к размеру диска

Ниже приведены формулы определения требований к дисковому пространству для хранения данных в IndustrialSQL Server. Эти формулы берут в расчет, что NTFS-сжатие файлов уменьшает реальный объем дискового пространства с 2:1 до 10:1. В худшем случае, используется фактор 2:1.

Система IndustrialSQL Server предназначена для работы на компьютере с файловой системой NTFS, которая является единственной файловой системой, поддерживаемой в промышленной среде. При использовании IndustrialSQL Server на компьютере с файловой системой FAT может произойти повреждение данных, если произойдет сбой системы, когда файлы все еще открыты.

---

**Внимание!** В циклической области хранения должно быть достаточно места на диске для содержания минимум двух полных архивных блоков, плюс свободное пространство для минимального объема циклической области хранения. Используйте InSQL Configure для просмотра или изменения значения минимального порога.

---

## Требования циклического хранения аналоговых данных

Каждое циклически сохраненное значение, записанное в базе данных IndustrialSQL Server, занимает около 4 байт пространства и плюс около 10% сверх того. Для оценки занимаемого дискового пространства за день для циклически сохраняемых аналоговых значений используется следующая формула:

Примерный объем использования диска =  $(4.4 * \text{количество тэгов}) * (60 / \text{скорость сохранения в секундах}) * 60 \text{ минут} * 24 \text{ часа} / (\text{процент NTFS-сжатия})$

Например, использование диска в день для 10000 тэгов, которые сохраняются с интервалом в 10 секунд, будет  $(4.4 * 10000) * (60 / 10) * 60 * 24 / 2$ , что составляет почти 190Мб в день.

## Требования дельта-хранения дискретных данных

Каждое значение, сохраненное в базе данных IndustrialSQL Server дельта-методом, занимает около 7 байт. Так как дельта-метод действует лишь при изменении значения, количество используемого пространства зависит от того, как часто меняются значения. Для оценки занимаемого дискового пространства в день для дискретных значений, сохраненных дельта-методом, используется следующая формула:

Требуемое дисковое пространство в день =  $(7 * \text{количество тэгов}) * (60 / \text{средняя скорость сохранения в секундах}) * 60 \text{ минут} * 24 \text{ часа} / (\text{процент NTFS-compression ratio})$

Например, использование дискового пространства в день для 10000 тэгов, которые в среднем изменяются каждые 60 секунд, будет  $(7 * 10000) * (60 / 60) * 60 * 24 / 2$ , что дает примерно 50 Мб в день.

## Требования дельта-хранения аналоговых данных

Каждое значение, сохраненное в базе данных IndustrialSQL Server дельта-методом, занимает около 10 байт. Так как дельта-метод действует лишь при изменении значения, количество используемого пространства зависит от того, как часто меняются значения. Для оценки занимаемого дискового пространства в день для аналоговых значений, сохраненных дельта-методом, используется следующая формула:

Требуемое дисковое пространство в день =  $(10 * \text{количество тэгов}) * (60 / \text{средняя скорость сохранения в секундах}) * 60 \text{ минут} * 24 \text{ часа} / (\text{процент NTFS-сжатия})$

Например, использование дискового пространства в день для 10000 тэгов, которые в среднем изменяются каждые 60 секунд, будет  $(10 * 10000) * (60 / 60) * 60 * 24 / 2$ , что составляет примерно 72 Мб в день.

---

**Примечание.** Аналоговое дельта-хранение поддерживается только в IndustrialSQL Server 7.0 и выше.

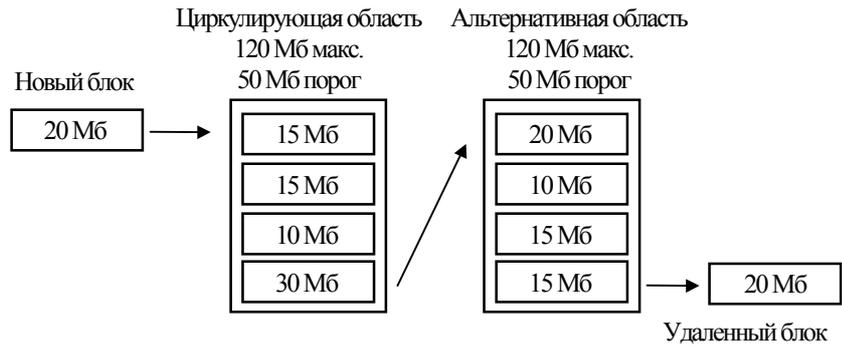
---

## Системная перезапись архивных блоков

Архивные блоки составляют таблицы расширения (или архивные таблицы, если используется OLE DB) базы данных IndustrialSQL Server и прогрессивно увеличиваются в размере по мере сбора и сохранения данных. Перед созданием нового блока система IndustrialSQL Server определяет количество оставшегося свободного пространства. Если места достаточно, создается новый блок и сохранение продолжается.

Если свободное место превышает определенный порог, система хранения удаляет старейший блок для освобождения места. Минимальный порог есть значение, используемое системой для определения необходимости удаления устаревшего блока. (Это значение хранится в таблице *StorageLocation*). Важно, чтобы администратор системы IndustrialSQL Server постоянно проверял свободное место на диске и резервировал устаревшие блоки во избежание их перезаписи.

Эта концепция показана на следующем примере.



---

**Примечание.** Данный пример специально сделан малым; типичные области хранения должны быть гораздо больше.

---

В данном примере архивный блок сохраняется в циклической области. Размер диска для циклического хранения равен 120 Мб. В дополнение к циклической области хранения, создана альтернативная область в 120 Мб. Для обеих областей минимальный порог равен 50 Мб. В общем, это означает, что полноценный размер пространства для хранения = 70 Мб.

Перед сохранением нового архивного блока система подсчитывает сумму размеров всех архивных блоков (включая и новый) в области хранения и определяет, достаточно ли свободного места для содержания всех блоков. Если нет, она перемещает достаточное количество архивных блоков из области хранения для получения требуемого количества свободного места и затем вставляет новый архивный блок. Как показано в примере, то же самое происходит и с альтернативной областью хранения. Архивный блок продолжает "перемещаться" по выбранным областям хранения, пока в конце концов не удаляется. Лучший способ предотвращения удаления данных — сжатие их при сохранении.

Если через какое-то время проверить свободное место в циклической области хранения, значение будет колебаться между значением порога и максимальным размером области с резкими возрастаниями при перемещении блока. Когда система перемещает блок(и), свободное пространство едва превысит порог перед возрастанием.

## Запуск системы хранения

Для запуска и остановки системы хранения используется InSQL Control. Подробнее см. в главе 5, "InSQL Control" в этом руководстве администратора.

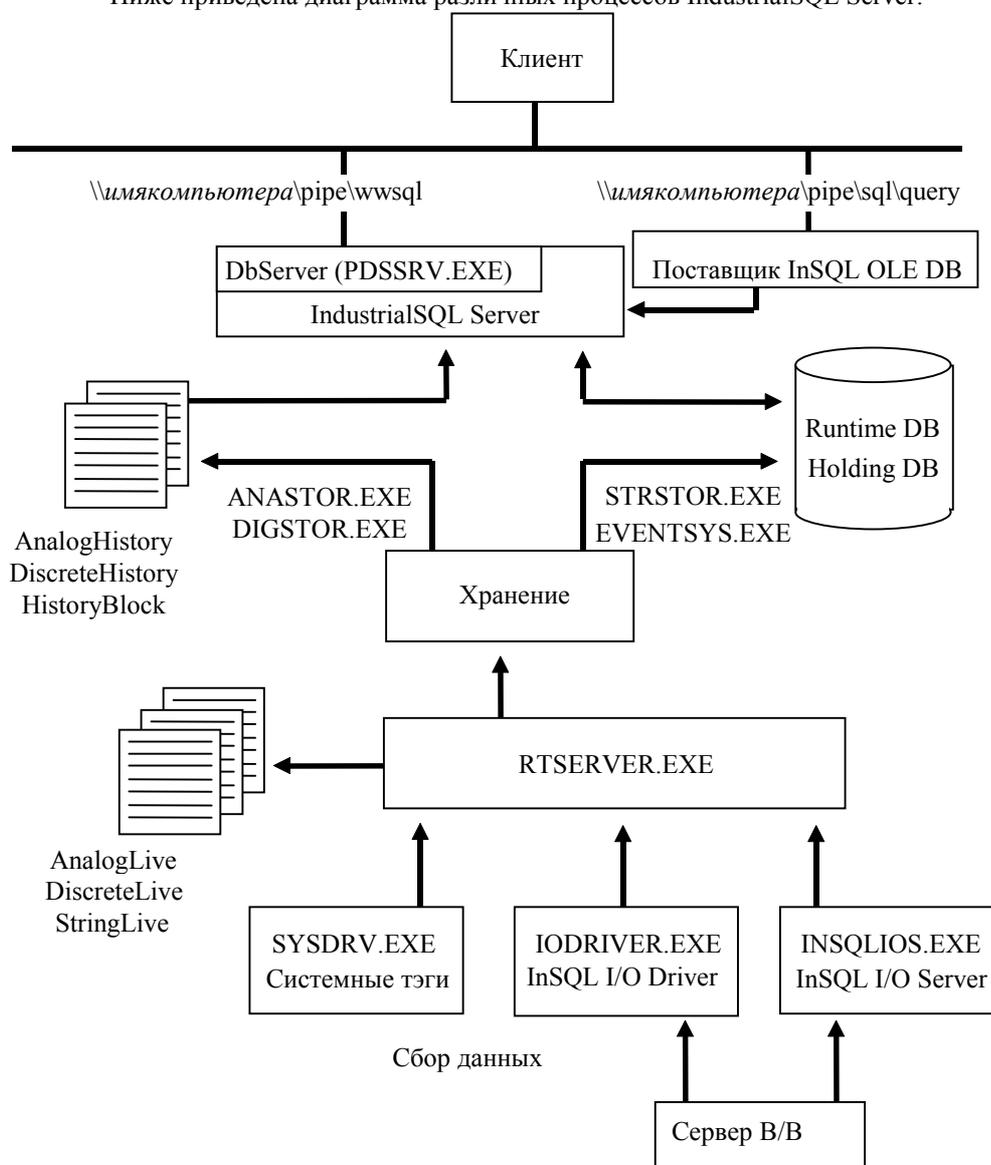
## Операции системы хранения

Система хранения состоит из ряда программ, которые выполняют конкретные задачи системы. При запуске системы хранения посредством InSQL Control каждое задание системы хранения запускается индивидуально. Задания системы хранения IndustrialSQL Server запускаются в заранее определенном порядке.

Задача	Порядок запуска	Описание
RTSERVER.EXE	1	Задача маршрута данных, инициализирующая процесс сохранения в активном образе.
MSFMERF.EXE	2	Дополнительный обработчик сообщений.
IODRIVER.EXE	3	Зависимый от системы драйвер. Читывает из базы данных IndustrialSQL Server и выполняет процесс инициализации активного образа.
SYSDRV.EXE	4	Драйвер системного монитора. Создает переменные состояния базовой системы и пересылает их через систему хранения IndustrialSQL Server.
DIGSTOR.EXE	5	Обработчик хранения дискретных данных.
ANASTOR.EXE	6	Обработчик хранения аналоговых данных.
PDSMGR.EXE	7	Распределитель дискового пространства.
STRSTOR.EXE	8	Обработчик хранения текстовых данных.
PDSSRV.EXE	9	Сервер выборки базы данных для Службы открытых данных (ODS). Обрабатывает запросы приложения-клиента. (Запросы, получающие доступ к данным через поставщика данных OLE DB, не используют PDSSRV.EXE).
INSQLIOS.EXE	10	Сервер ввода-вывода реального времени. Используется для получения текущих значений элементов для заданных тем.
EVENTSYS.EXE	11	Система управления событиями. На основе конфигурации событий фиксирует события и выполняет связанные с ними действия.

При запуске системы хранения IndustrialSQL Server через InSQL Control можно просматривать состояние загрузки каждой задачи в основном окне.

Ниже приведена диаграмма различных процессов IndustrialSQL Server:



**Примечание.** Запросы, получающие доступ к данным через поставщика OLE DB, не используют PDSSRV.EXE.

## Службы IndustrialSQL Server

Следующие задачи запускаются как службы.

Задача	Имя службы	Описание
INSQLCNTL.EXE	InSQL Control	Программа InSQL Control, используемая для запуска и остановки системы.
PDSSRV.EXE	InSQL DbServer	Сервер базы данных для служб открытых данных (ODS). Этот процесс может быть запущен и остановлен независимо от остальной системы. Если эта служба остановлена, приложения-клиенты не смогут получить доступ к IndustrialSQL Server. Тем не менее, сохранение данных продолжится.
EVENTSYS.EXE	InSQL Event System	На основе конфигурации событий фиксирует события и выполняет связанные с ними действия.
IODRIVER.EXE	InSQL IO Driver	Считывает из базы данных IndustrialSQL Server и выполняет процесс инициализации активного объекта. Позволяет посылать данные с устройств данных (серверов ввода-вывода) в систему сбора.
INSQLIOS.EXE	InSQL I/O Server	Поставляет последние значения приложениям-клиентам, таким как InTouch.

## Резервирование архивных данных

Будучи реляционной и быстродействующей СУБД, сервер может сохранять большой объем данных за длительное время прямо на жестком диске, обеспечивая высокоскоростную выборку сохраненных данных через поддерживаемые сети. Когда создается архивный блок, может возникнуть необходимость перемещения некоторых архивных блоков на устройства длительного хранения, чтобы освободить место для сохранения новых архивных блоков.

Устройства длительного хранения, используемые для резервирования архивных блоков могут включать в себя магнитные DAT-кассеты или оптические устройства.

Подробнее см. в главе 9, "Резервирование данных и баз данных" в этом руководстве администратора.

## Поддержка перевода часов на летнее/зимнее время

IndustrialSQL Server поддерживает последовательное сохранение и выборку данных в периоды перехода на летнее/зимнее время. Действия системы зависят от того, переводятся ли часы назад или же вперед.

**Внимание!** Не используйте блоки длиной менее 3 часов в периоды смены летнего/зимнего времени. В этом случае IndustrialSQL Server не будет автоматически определять смену времени и данные будут потеряны. Стандартная длина блока в этом случае равняется 24 часам, что вполне приемлемо.

Так как IndustrialSQL Server настроен на длину блока равную 3 часам или более, перевод часов будет проведен как описано в следующих разделах.

**Примечание.** Microsoft SQL Server кэширует ранее полученные данные для последующего использования во время текущего сеанса работы. Если каким-то образом архивные блоки были увеличены (например, при переводе часов на зимнее/летнее время или при копировании архивных блоков), клиент должен разъединиться и вновь соединиться с сервером для обновления и получения новейших данных.

### Перевод времени на час вперед

Для этого перехода локальное время переводится на час вперед, создавая в IndustrialSQL Server часовой разрыв между сохраненными и собранными данными. Это не требует каких-то особенных действий со стороны IndustrialSQL Server, так как все сохраненные данные все так же доступны пользователю. Следующий пример показывает смещение времени по Тихоокеанскому побережью 4 апреля 1999 г. при условии, что IndustrialSQL Server настроен на 24-часовой блок.

	3 апреля	4 апреля	
	11:59PM	12:00AM	2:00AM
			3:00AM
Блок предыдущего дня			
Блок 1		Блок 1	

## Перевод времени на час назад

Для этого перехода локальное время перемещается на час назад, что вызывает в IndustrialSQL Server наложение одного часа в сохраняемых и собираемых данных. Это требует специальных действий по отношению к блокам данных от IndustrialSQL Server для предотвращения перезаписи в существующем блоке данных. В следующем примере показано смещение стандартного времени на Тихоокеанском побережье 31 Октября 1999 г. при условии, что IndustrialSQL Server настроен на 24-часовую длину блока.

	октябрь 30	октябрь 31			
	11:59PM	12:00AM	1:00AM	2:00AM	3:00AM
Блок предыдущего дня					
Блок 1		Блок 1			
Блок 2			Блок 2		
Блок 3			Блок 3		Блок 4 4 (создан вручную)

Заметьте, что существует период наложения данных в Блоке 1 и Блоке 3 на один час. По умолчанию запрос на выборку данных в период между 1:00 a.m. и 2:00 a.m. 31 октября 1999 г. выдаст данные из Блока 2. Блок 3 будет продолжать заполняться до следующей запланированной смены блока (стандарт -- 24 часа).

Для доступа к данным между 1:00 a.m. и 2:00 a.m. из Блока 3 возможно понадобится отредактировать архивный блок (.INF файл) для Блока 3. Однако файл блока нельзя редактировать, если файл все еще записывается. Для того, чтобы отредактировать файл Блока 3, следует создать вручную новый архивный блок (Блок 4 в диаграмме), используя InSQL Control. Когда создан новый блок, можно редактировать Блок 3.

[Подробнее об использовании InSQL Control см. в главе 5, "InSQL Control".](#)

## Доступ к наложенному блоку данных

Если вы хотите получить доступ к первому часу данных Блока 3, но не к Блоку 2, может потребоваться внести изменения в файл BLOCK.INF, расположенный в связанном с Блоком 3 каталоге в циклической области хранения IndustrialSQL Server.

Файл BLOCK.INF — это текстовый файл, содержащий информацию о содержимом блока данных. Он может быть просмотрен любым текстовым редактором, таким как Блокнот.

---

**Внимание!** Обычно не требуется править эти файлы, подобные действия могут привести к потере возможности получать данные из связанного блока данных.

---

Для примера ниже приведен типичный файл BLOCK.INF для Блока 3:

```
Данные InSQL: A991031D
1999-10-31 02:00:00
1999-11-01 00:00:00
303
28800
:
```

Правка выделенного жирным шрифтом позволит получать данные из Блока 3 в период наложения.

```
Данные InSQL: A991031D
1999-10-31 01:00:00
1999-11-01 00:00:00
303
28800
:
```

---

**Примечание.** Отредактировать и сохранить файл BLOCK.INF для Блока 3 можно только тогда, когда он не является текущим записываемым блоком. Другими словами, IndustrialSQL Server увеличил число блоков в результате перезапуска системы, создания вручную или как результат запланированной смены архивного блока.

---

## Настройка дельта-хранения для быстро изменяющихся тэгов

В этом разделе описано, как следует настроить систему хранения, чтобы оптимизировать дельта-метод для быстро изменяющихся тэгов.

### Как работает дельта-хранение

При запуске IndustrialSQL Server система хранения не может определить, сколько изменений будет получено для каждого тэга и не настраивается соответствующим образом. Также система хранения не может записывать каждое изменение на диск немедленно, поскольку устройство ввода-вывода может потреблять большее количество времени и ресурсов. Для решения этих проблем система использует комбинацию снимков и буферов значений тэгов. Снимок содержит копию всех дельта-сохраненных значений тэгов в конкретный момент времени. Буферы для каждого тэга в снимке содержат последующие изменения значений, которые применяются как динамические массивы и могут расти почти бесконечно.

Дельта-снимки и их динамические массивы хранятся в файлах .SDT в папке архивных блоков. Существует файл снимка для каждого типа тэга, отличающегося по байтовому размеру. Например, тэги целого и длинного типа являются 4-байтовыми и они будут сохраняться в 4-байтовых файлах снимков. Существует 4 файла снимков: 1-байтовый, 2-байтовый, 4-байтовый и 8-байтовый.

Когда система запущена впервые, первый снимок создается в файле .SDT. Этот снимок содержит исходные значения для дельта-сохраненных тэгов. Новые дельта-значения для каждого тэга затем добавляются в динамический массив этого тэга. Каждые 30 секунд снимок обновляется с учетом *всех* изменений, произошедших со времени создания снимка.

Когда размер текущего снимка достигает определенной границы (по умолчанию 2 Мб) или когда проходит час, создается второй снимок и запускается новый динамический массив для каждого тэга. Когда достигнут предел величины или прошел час, создается третий снимок и т.д. Например:

<b>Snapshot 0</b>
Deltas for Tag A
Deltas for Tag B
...
Deltas for Tag Z
<b>Snapshot 1</b>
Etc.
<b>Snapshot n</b>

## Дельта-хранение и быстро изменяющиеся тэги

В случае с быстро изменяющимися тэгами будет получено большое количество значений в снимковых динамических массивах за короткий промежуток времени (например, за 10-30 минут накопления данных). Когда система готова обновить снимок (каждые 30 секунд), она должна пройти через *каждое* изменившееся значение в *каждом* динамическом массиве для *каждого* тэга с начала снимка. Поскольку система хранения работает как задача с высоким приоритетом, операционная система старается найти достаточно ресурсов для интенсивной работы. В результате остальным подсистемам не хватает ресурсов для нормального функционирования.

Это не происходит с медленно изменяющимися тэгами, с тэгами, сохраняемыми со скоростью от 1 до 5 секунд или с циклично сохраняемыми тэгами. Снимок этих типов тэгов легко управляется во временной рамке от 10 до 30 минут.

## Изменение размера снимков для быстро изменяющихся тэгов

Для уменьшения загрузки системы при обработке быстро изменяющихся тэгов новый снимок должен создаваться чаще и с меньшим количеством элементов в динамических массивах. Для этой цели используются следующие ключи реестра:

```
HKEY_LOCAL_MACHINE
  \SOFTWARE
    \Wonderware
      \InSQL
        \Storage
          \1ByteSnapshotSize
          \2ByteSnapshotSize
          \4ByteSnapshotSize
          \8ByteSnapshotSize
```

Заметьте, что снимки группируются по размеру, а не по типу хранения. Ключи реестра имеют значения в формате DWORD, указывающие размер в байтах для каждого типа снимка.

Правильный размер снимка зависит от загрузки системы. Обычно, наилучшее время около 10 минут. Для более точного подсчета размера снимка в случае быстро изменяющегося системного тэга используется следующая формула.

Стандарты формулы, в байтах:

- TS** – размер типа (истинный объем хранения)
- SVS** – размер сохраняемого значения (TS + 3)
- SHS** – размер заголовка снимка (число тэгов \* (TS + 3))
- SPS** – число выборок в секунду
- NTn** – количество тэгов в указанном диапазоне SPS

Формула подсчета размера снимка:

$$\text{Размер} = (NT1 * (10*60*SPS) + NT2 * (10*60*SPS) \dots NTn * (10*60*SPS)) SVS + SHS$$

Следующий пример показывает, как использовать формулу для подсчета размера 4-байтовых тэгов (целых, длинных и т.п.):

$$\begin{aligned} TS &= 4 \text{ байта} \\ SVS &= 4 + 3 = 7 \\ SHS &= (35 * (4 + 3)) = 35 * 7 = 259 \\ SPS &= 20 \text{ (5 тэгов, изменяемых каждые 50 мсек)} \\ SPS &= 1 \text{ (30 тэгов, изменяемых каждую 1 секунду)} \\ NT1 &= 5 \\ NT2 &= 30 \end{aligned}$$

Соответственно:

$$\text{Размер снимка} = (5*(600*20)+30*(600 * 1) ) 7+ (110*7)$$

$$\text{Размер снимка} = 546770 \sim 0.5M$$

Таким образом значение *4ByteSnapdhotSize* устанавливается на 546770 байт.

# InSQL Control

В этой главе рассказывается о программе InSQL Control, позволяющей контролировать и следить за компонентами системы IndustrialSQL Server с одного пользовательского интерфейса. InSQL Control позволяет запускать и останавливать систему хранения и следить за переменными, связанными с драйверами выборки данных.

Подробнее о системе хранения см. в главе 4, "Управление системой хранения" этого руководства администратора.

## Содержание

- О программе InSQL Control
- Действие лицензии при запуске системы
- Запуск InSQL Control
- Запуск системы
- Остановка системы
- Отслеживание сбора данных
- Изменение режима просмотра
- Сброс счетчика ошибок
- Создание нового архивного блока
- Изменение учетной записи для службы
- Изменение учетной записи для SQL Server

## О программе InSQL Control

Чтобы убедиться, что система IndustrialSQL Server работает с максимальной эффективностью, важно отслеживать работу компонентов системы хранения и системы сбора данных IndustrialSQL Server. InSQL Control — это приложение, используемое для запуска, остановки и отслеживания работы системы IndustrialSQL Server. С помощью InSQL Control можно оценить общую скорость работы драйверов, а также скорость обработки данных каждым отдельным драйвером.

---

**Внимание!** Если стандартная учетная запись **wwAdmin** была изменена, InSQL Control не запустится. [Подробнее см. в главе 13, "Безопасность"](#). Если сервер базы данных InSQL (PDSSRV.EXE) работает независимо от остальной системы, следует удостовериться, что больше никакие части системы не работают, включая InSQL Control.

---

## Действие лицензии при запуске системы

При запуске IndustrialSQL Server проверяется наличие в системе действующей лицензии Wonderware. Если файла лицензии нет, IndustrialSQL Server и Microsoft SQL Server (если установлен с установочного CD IndustrialSQL Server) будет запущен в режиме демонстрации на два часа, после чего выключится. Если IndustrialSQL Server установлен для использования с уже присутствующей системой Microsoft SQL Server, приобретенной отдельно от IndustrialSQL Server, данная версия Microsoft SQL Server продолжит работу.

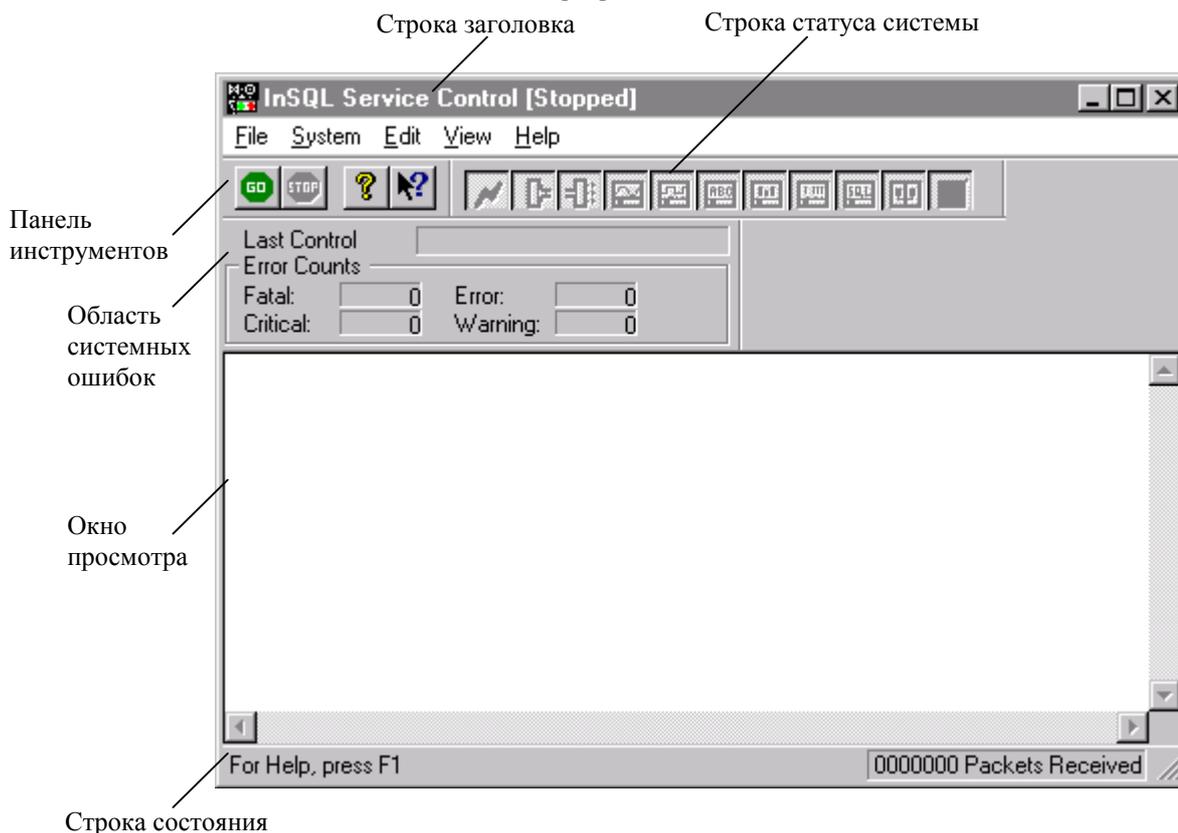
Когда IndustrialSQL Server настроен на автоматический запуск, в журнал ошибок записывается сообщение, если лицензия отсутствует или просрочена. Если IndustrialSQL Server настроен на ручной запуск, выдается запрос на подтверждение запуска в демо-режиме.

## Запуск InSQL Control

### ➤ Как запустить программу InSQL Control:

1. Из меню **Пуск** выберите группу программ FactorySuite, затем в группе IndustrialSQL Server выберите InSQL Control.

Появится главное окно программы.



Главное окно InSQL Control состоит из строки заголовка, панели инструментов, строки статуса системы, области системных ошибок, окна просмотра и строки состояния.

### Строка заголовка

Строка заголовка показывает текущее состояние службы.

## Панель инструментов

По умолчанию панель инструментов InSQL Control находится под строкой меню. Эта панель содержит кнопки для доступа к часто используемым командам меню:

Кнопка	Действие
	Запустить систему IndustrialSQL Server.
	Остановить систему.
	Показать окно <b>О программе InSQL Service Control</b> .
	Показать контекстно-зависимую справку.

## Строка состояния системы

Справа от панели инструментов находятся индикаторы состояния процессов и "здоровья" системы. При запуске каждого системного процесса загорается соответствующий индикатор состояния. Каждый из этих индикаторов описан в приведенной ниже таблице:

Индикатор состояния	Указывает
	Состояние сбора данных.
	Состояние работы драйвера ввода-вывода.
	Состояние работы сервера реального времени.
	Состояние аналогового хранения.
	Состояние дискретного хранения.
	Состояние текстового хранения.
	Состояние дополнительного пакетного сбора.
	Состояние системы управления событиями.
	Состояние сервера базы данных.
	Статус сервера реального времени ввода-вывода IndustrialSQL.
	Состояние "здоровья" системы. При критической или фатальной ошибке этот индикатор превращается в красный "X".

## Область системных ошибок

Эта область InSQL Control отображает коды ошибок в работе системы и их количество. Окно **Last Control Error** показывает все ошибки, произошедшие со времени запуска. Количество ошибок автоматически сбрасывается на ноль при запуске и выключении системы.

## Окно просмотра

Эта область of InSQL Control используется для просмотра состояния каждого процесса системы хранения при запуске (просмотр сообщений) или состояния всех тем данных (просмотр тем). Команда **Copy** в меню **Edit** позволяет копировать содержимое окна просмотра в буфер обмена Windows.

По умолчанию InSQL Control запускается в режиме просмотра сообщений. [Подробнее см. в разделе "Изменение режима просмотра" далее в этой главе.](#)

## Строка состояния

По умолчанию строка состояния появляется внизу главного окна InSQL Control. Она отображает информацию о командах меню, а также число системных драйверов IndustrialSQL и пакетов данных, полученных драйвером ввода-вывода:



Системный драйвер (SYSDRV.EXE) показывает внутренние переменные "здоровья" системы IndustrialSQL Server.

При наведении указателя мыши на меню InSQL Control, в левой области строки состояния будет показано описание этой команды меню. Если после просмотра описания команды панели инструментов вы передумали выполнять ее, просто отпустите кнопку мыши, когда указатель находится вне кнопки.

## Отображение или скрытие программных элементов

Можно отобразить любые программные элементы InSQL Control для использования при выполнении команд или скрыть их для получения большей области просмотра. Программные элементы включают в себя панели инструментов, строку состояния, строку статуса системы, область ошибок и т.д.

- **Как отобразить программный элемент:**
  1. Выберите команду меню **View**.
  2. Выберите нужное меню (например, меню **Toolbar**), чтобы рядом с ним появилась галочка.
  3. Программный элемент будет показан в приложении.
  
- **Как скрыть программный элемент:**
  1. Выберите команду меню **View**.
  2. Выберите нужное меню (например, меню **Toolbar**), чтобы галочка исчезла.
  3. Программный элемент будет скрыт в приложении.

## Запуск системы

При запуске системы все процессы последовательно запускаются и начинают выполнять каждый свою задачу. Некоторые операции, выполняемые во время загрузки:

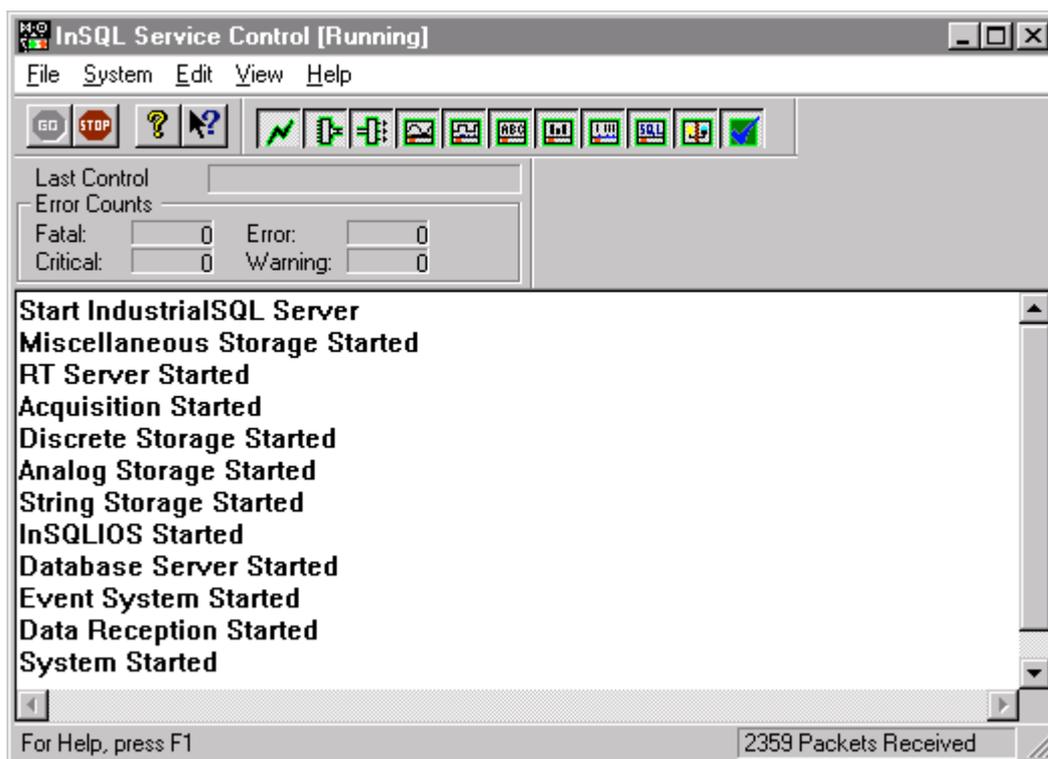
- Проверка информации запуска в сервере SQL и в реестре.
- Запуск сервера сбора данных.
- Создание нового архивного блока на жестком диске для хранения данных.
- Установка соединения с источниками данных, такими как драйверы и серверы ввода-вывода.
- Начало сохранения данных в новом архивном блоке.

Все процессы будут запущены как службы.

## Выполнение процедуры запуска

### ➤ Как запустить систему:

1. В меню **Пуск** на Панели задач Windows выберите программную группу FactorySuite, затем в подгруппе IndustrialSQL Server выберите InSQL Control.
2. В меню **System** выберите **Start**. Или нажмите кнопку . При загрузке процессов системы хранения в главном окне будут показаны соответствующие сообщения.
3. По ходу запуска системных процессов будут загораться индикаторы состояния, отображая состояние выполнения этих процессов.



Когда все процессы успешно запущены и начинается активное сохранение данных, счетчик пакетов в строке состояния начинает показывать данные. Счетчик пакетов показывает количество пакетов данных, полученных системой хранения от уровней сбора данных IndustrialSQL Server.

## Системные процессы

Система состоит из нескольких индивидуальных программ, которые исполняют свои функции в системе. При запуске системы через InSQL Control каждая задача системы запускается индивидуально. Задачи IndustrialSQL Server запускаются в заранее определенном и фиксированном порядке.

Задача	Порядок запуска	Описание
MSFMERF.EXE	1	Дополнительный обработчик сообщений.
RTSERVER.EXE	2	Задача маршрутизации данных, включающая процесс активного образа.
SYSDRV.EXE	3	Драйвер состояния системы. Создает основные переменные системного "здоровья" и передает их через систему хранения IndustrialSQL Server.
IODRIVER.EXE	4	Драйвер сбора данных. Получает данные от серверов ввода-вывода.
DIGSTOR.EXE	5	Обработчик хранения дискретных данных.
STRSTOR.EXE	6	Обработчик хранения текстовых данных.
ANASTOR.EXE	7	Обработчик хранения аналоговых

		данных.
PDSMGR.EXE	8	Диспетчер дискового пространства.
INSQLIOS.EXE	9	Сервер ввода-вывода IndustrialSQL
PDSSRV.EXE	10	Сервер базы данных.
EVENTSYS.EXE	11	Система управления событиями

При запуске системы IndustrialSQL Server посредством InSQL Control в главном окне можно видеть состояние загрузки каждой задачи. Все процессы будут запущены как службы.

Подробнее о хранении данных см. в главе 4, "Хранение данных".

Подробнее о сервере ввода-вывода IndustrialSQL см. в главе 10, "Настройка серверов ввода-вывода".

Подробнее о системе управления событиями см. в главе 7, "Система управления событиями".

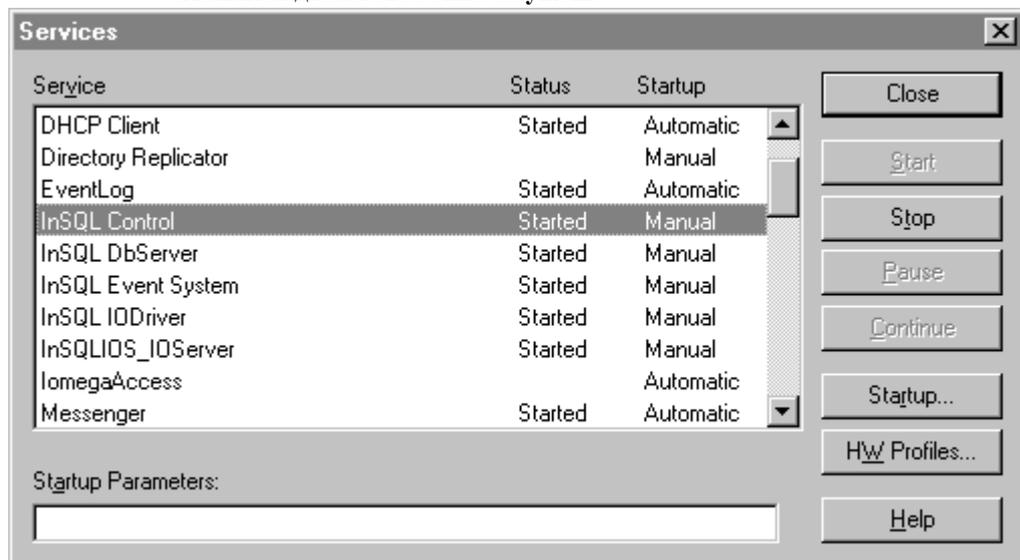
## Настройка автоматического запуска

Компьютер можно настроить на автоматический запуск службы InSQL Control при загрузке системы. Только одна служба InSQL Control может быть запущена в системе. Изменение настройки запуска службы InSQL Control на автоматическую не будет иметь эффекта до перезагрузки системы.

### ➤ Как настроить автозапуск:

1. В панели управления дважды нажмите на **Службы**.

Появится диалоговое окно **Службы**.



2. В списке служб выберите InSQL Control.
3. Выберите пункт **Загрузка**. Появится диалоговое окно **Служба**.



4. Из списка **Тип загрузки** выберите **Автоматическая**.
5. Нажмите **ОК**.

## Остановка системы

После остановки системы никаких данных на диск сохраняться не будет. Рекомендуется выключать систему таким образом, как описано в этом разделе.

### ➤ Как остановить систему:

1. Из меню **System** выберите **Stop**. Или нажмите кнопку  на панели инструментов.
2. В то время, как все системные процессы выключаются, в окне просмотра будут появляться сообщения и соответствующие кнопки состояния в панели инструментов станут недоступными.
3. После остановки системы можно закрыть InSQL Control.

## Отслеживание сбора данных

С помощью программы InSQL Control можно отслеживать состояние данных, полученных от каждого драйвера, или общее состояние данных, полученных от всех драйверов. Эти драйверы, также называемыми источниками данных, включают в себя:

- Серверы ввода-вывода.
- Системный драйвер, показывающий переменные "здоровья" IndustrialSQL Server.

Типы информации о состоянии сервера, которые могут быть просмотрены, объясняются в приведенной ниже таблице:

<b>Индикатор состояния</b>	<b>Описание</b>
CPS	Значение счетчика символов в секунду.
Bytes	Значение счетчика полученных байт.
Seconds	Количество секунд со момента запуска.
Packets	Количество пакетов, полученных с момента запуска.
Nacks	Количество переданных негативных подтверждений.
Restarts	Количество перезапусков драйвера.

## Отслеживание общего счетчика пакетов

Общий счетчик пакетов показывает текущее количество пакетов, полученных всей системой сбора данных. Общий счетчик пакетов автоматически отображается с правой стороны строки состояния. Когда система работает, общий счетчик пакетов будет постоянно меняться по ходу перемещения собранных данных в систему хранения.

## Отслеживание сбора данных

В программе InSQL Control можно просмотреть сводку информации по драйверам системы хранения. Эта сводка включает в себя подсчеты передачи и сбора данных для всех определенных в системе серверов данных.

➤ **Как просмотреть счетчик собранных данных:**

1. Из меню **View** выберите **Acquisition Totals**.

Появится диалоговое окно **Driver Status**:



Это диалоговое окно показывает общее количество символов в секунду (CPS), байт, пакетов, негативных подтверждений и перезагрузок для всех драйверов системы.

Подробнее см. в разделе "Отслеживание сбора данных" ранее в этой главе.

2. Нажмите **OK**.

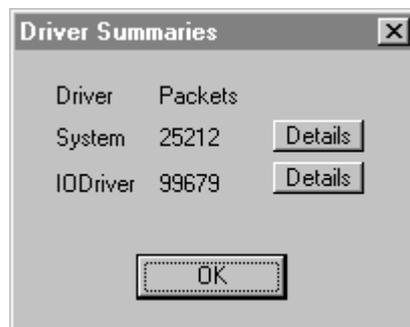
## Отслеживание деталей сбора данных

В программе InSQL Control можно просмотреть статистику сбора данных для каждого определенного сервера данных или драйвера данных. Это позволяет отслеживать процесс работы индивидуальных драйверов данных в сравнении со старыми архивами или с другим драйвером данных. Используя InSQL Control, можно легко отслеживать счетчик пакетов сразу для всех определенных драйверов данных, или просматривать детали каждого сервера данных.

➤ **Как просмотреть детали сбора для каждого драйвера данных:**

1. В меню **View** выберите **Acquisition Details**.

Появится диалоговое окно **Driver Summaries**.



Это диалоговое окно показывает общее количество пакетов, полученных для каждого определенного драйвера данных.

2. Для просмотра деталей конкретного драйвера данных нажмите кнопку **Details** справа от имени драйвера.

Появится диалоговое окно **Driver Status**, показывающее состояние передачи и сбора данных для индивидуального драйвера.



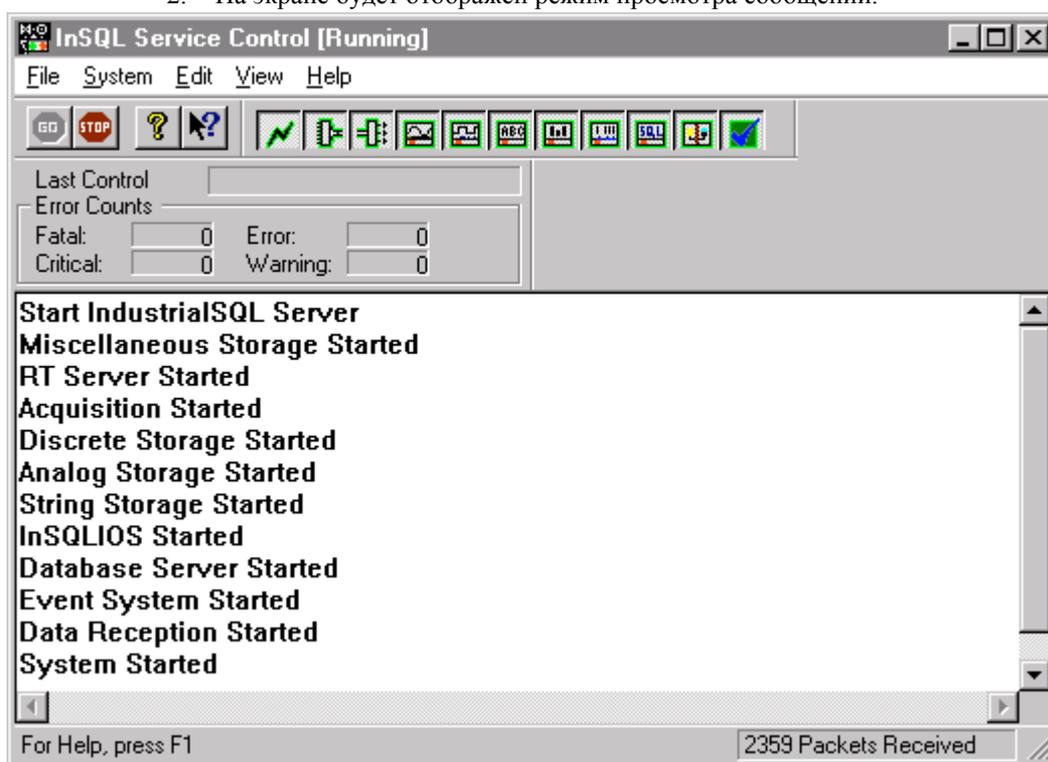
3. Для закрытия этого окна нажмите **OK**.
4. Для закрытия диалогового окна **Driver Summaries** нажмите **OK**. (Окно **Driver Summaries** можно также закрыть, когда окно **Driver Status** все открыто).

## Изменение режима просмотра

Окно просмотра в InSQL Control имеет два режима: просмотр сообщений и просмотр тем. По умолчанию InSQL Control запускается в режиме просмотра сообщений. В этом режиме можно посмотреть состояние всех процессов.

➤ **Как включить режим просмотра сообщений:**

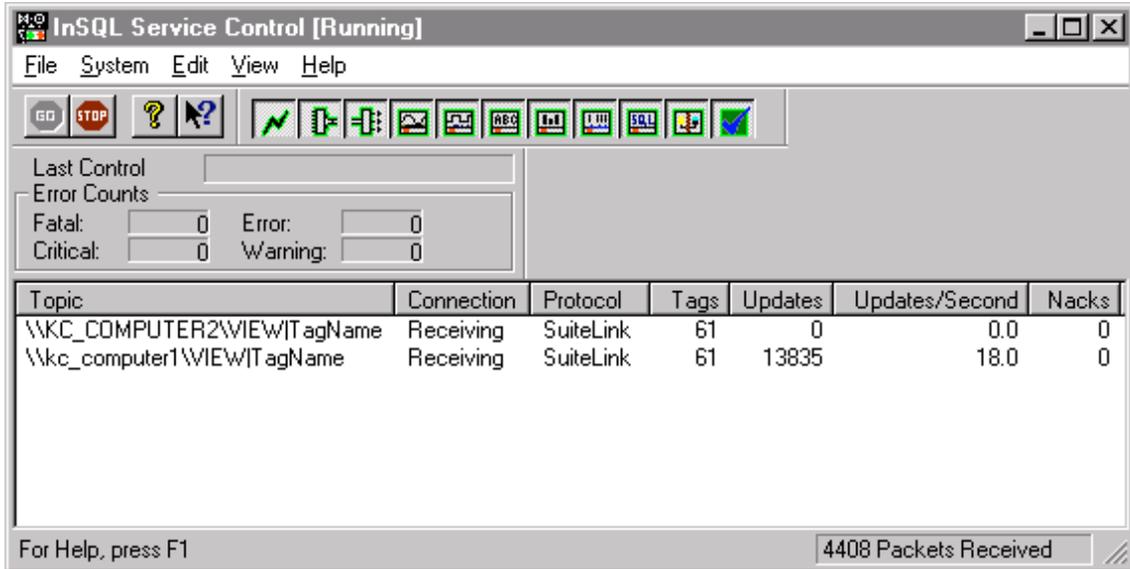
1. Из меню **View** выберите **Messages**.
2. На экране будет отображен режим просмотра сообщений.



Режим просмотра тем показывает текущие темы и их состояния, как это предусмотрено драйвером данных (IODRIVER.EXE).

➤ **Как включить режим просмотра тем:**

1. Из меню **View** выберите **Topics**.
2. На экране будет отображен режим просмотра тем.



Следующая таблица описывает столбцы режима просмотра тем:

Столбец	Описание
Topic	Имя темы.
Connection	Состояние соединения. Возможные состояния: <b>Error, Idle, Connecting, Connected, Receiving, Disconnect, Overload.</b>
Protocol	Протокол, используемый для темы. <b>DDE</b> или <b>SuiteLink</b> .
Tags	Количество тэгов в теме.
Updates	Количество значений, обновленных в теме с тех пор, как с этой темой последний раз устанавливалось соединение.
Updates/Second	Количество значений, обновленных за 1 секунду.
Nacks	Количество негативных подтверждений.

## Сброс счетчика ошибок

Счетчик ошибок автоматически сбрасывается на ноль при загрузке или выключении системы. Можно в любой момент сбросить счетчик ошибок обратно на ноль.

- Для сброса счетчиков в меню **System** выберите **Reset Error Counts**.

---

**Примечание.** Предупреждающего сообщения выдано не будет. Тем не менее, результат изменения счетчика можно увидеть в приложении-клиенте. Информационное сообщение также записывается в журнал ошибок.

---

## Создание нового архивного блока

IndustrialSQL Server сохраняет полученные данные на диск в подкаталоги, называемые архивными блоками. Архивные блоки создаются, когда система запускается или после окончания заранее определенного и указанного в реестре интервала времени

Можно вручную предотвратить запись данных в текущий архивный блок и создать новый. Этот процесс занимает около 10 минут, но потери данных не происходит. Система должна работать уже минимум 10 минут после запуска, прежде чем станет возможным создание нового архивного блока.

➤ **Как создать новый архивный блок вручную:**

- В меню **System** выберите **Start New Data Block**.

Создать новый архивный блок можно также при помощи расширенной хранимой процедуры **xp\_NewHistoryBlock**.

Подробнее о расширенной хранимой процедуре **xp\_NewHistoryBlock** см. в *Справочном руководстве IndustrialSQL Server*.

## Изменение учетной записи для службы

Все компоненты FactorySuite, запускаемые как службы, используют учетную запись пользователя "Master". Эта учетная запись пользователя FactorySuite шифруется и сохраняется в реестре под следующим ключом:

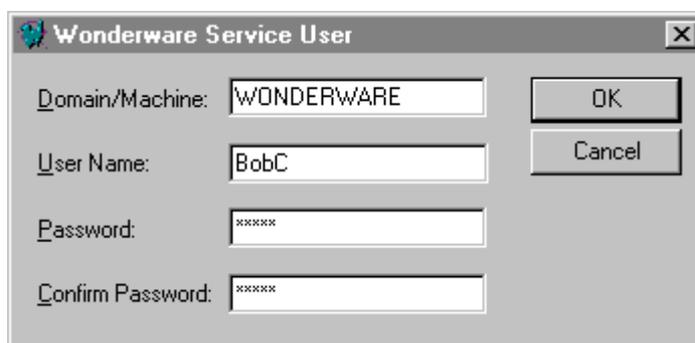
HKEY\_LOCAL\_MACHINE\Software\Wonderware\FS2000\Common\Security

Изменить учетную запись пользователя FactorySuite можно из InSQL Control. Подробнее см. в *Руководстве администратора системы FactorySuite*.

➤ **Как изменить учетную запись пользователя FactorySuite:**

1. Из меню **File** выберите **Set Service User Login**.

Появится диалоговое окно **Wonderware Service User**.



2. Введите параметры учетной записи пользователя, у которого будут права на запуск и остановку служб.

**Domain**

Имя домена, в котором будет проверяться учетная запись пользователя.

**Username**

Имя пользователя для данной учетной записи.

**Password**

Пароль для данной учетной записи.

**Confirm Password**

(повторный ввод пароля)

3. Нажмите **OK**.

---

**Примечание.** Учетную запись пользователя FactorySuite можно также изменить из программы Wonderware Service User. Подробнее см. в *Руководстве администратора системы FactorySuite*.

---

## Изменение учетной записи для SQL Server

Во время загрузки служба контроля IndustrialSQL проверяет реестр на предмет определения имени пароля и пользователя, которые она должна использовать для доступа к Microsoft SQL Server. Учетная запись для SQL Server шифруется и сохраняется в реестре под следующим ключом:

HKEY\_LOCAL\_MACHINE\Software\Wonderware\FS2000\Common\Security

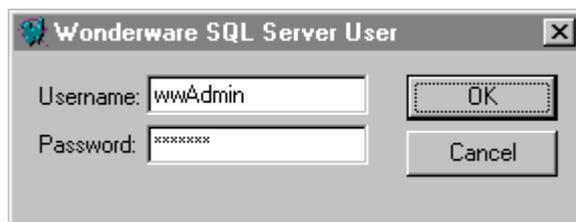
Если изменить имя пользователя или пароль "wwAdmin" без обновления ключа реестра для учетной записи "wwAdmin", InSQL Control не сможет получить доступ к SQL Server для запуска системы.

Установки реестра можно изменить при помощи InSQL Control.

➤ **Как изменить имя пользователя или пароля wwAdmin:**

1. В меню **File** выберите **Set SQL Server Login**.

Появится диалоговое окно **Wonderware SQL Server User**.



2. В поле **Username** введите новое имя пользователя для учетной записи.
3. В окне **Password** введите новый пароль для учетной записи.
4. Нажмите **OK**.

Зашифрованная учетная информация будет изменена.

# Выборка данных

В этой главе описаны аспекты выборки данных из базы IndustrialSQL Server. Рассматривается выборка данных через службы открытых данных (ODS) сервера IndustrialSQL Server и через поставщика данных OLE DB IndustrialSQL Server.

## Содержание

- Типы выборки данных
- Выборка через службы открытых данных (ODS)
- Выборка через поставщика данных OLE DB
- Сводка ODS и OLE DB

## Типы выборки данных

IndustrialSQL Server поддерживает два типа выборки данных: циклическое и дельта.

### Циклическая выборка

Циклической называется выборка сохраненных данных за заданный период времени, основанная на указанном разрешении и не зависящая от того, изменилось ли значение тэга (тэгов). Для управления разрешением выборки (интервала времени) могут использоваться расширения временного домена счетчика строк и разрешения.

Подробнее см. в главе 14, "Выборка данных", *Справочного руководства IndustrialSQL Server*.

### Дельта-выборка

Дельта-выборка или выборка методом исключения — это выборка измененных значений тэга за заданный период времени. То есть, дублируемые значения не выдаются. Только расширения временного домена счетчика строк могут быть применены к данным, выбираемым этим методом.

Подробнее см. в главе 14, "Выборка данных", *Справочного руководства IndustrialSQL Server*.

## Выборка через службы открытых данных (ODS)

Службы открытых данных (ODS) — это интерфейс программирования (API) между сетевыми библиотеками сервера и приложениями сервера. Службы ODS позволяют приложениям-клинтам получать доступ к данным, которые не хранятся физически в Microsoft SQL Server, к которому они присоединяются. Клиент видит данные так, как будто они располагаются в файле базы данных Microsoft SQL Server. В случае с IndustrialSQL Server, эти данные хранятся в файлах архивных блоков.

В версиях меньше 7.1 сервер базы данных IndustrialSQL был включен лишь как приложение ODS. Помимо поддержки этой функции, IndustrialSQL Server расширен для поддержки новых технологий OLE DB.

## Архитектура ODS IndustrialSQL Server

Шлюзовые серверы ODS, как сервер базы данных IndustrialSQL, имеют возможность выдавать данные клиентам Microsoft SQL Server так, как будто эти данные хранятся в таблице SQL Server. Со стороны клиента эти "таблицы расширения" (термин Wonderware, обозначающий данные, содержащиеся в архивных блоках IndustrialSQL Server) работают как любая другая таблица SQL Server и запрос в нее подается тем же образом.

Сервер базы данных IndustrialSQL (PDSSRV.EXE) перехватывает все запросы клиента на данные до того, как они достигнут Microsoft SQL Server, а если запрос идет на выборку данных из архивных блоков IndustrialSQL Server, сервер базы данных обращается к системе сбора IndustrialSQL Server, получает данные и представляет их пользователю в виде обычной таблицы SQL Server. Запросы, не касающиеся архивных блоков, проходят в SQL Server неизменными для стандартной обработки.

Диаграмму архитектуры ODS IndustrialSQL и OLE DB см. в разделе "Сводка ODS и OLE DB" далее в этой главе.

Версии IndustrialSQL ранее 7.1 становились стандартом установки соединения для SQL Server, что делало их серверами по умолчанию и обеспечивало альтернативную связь с Microsoft SQL Server. Версия же 7.1 оставляет сервером по умолчанию Microsoft SQL Server и обеспечивает альтернативное соединение с сервером базы данных IndustrialSQL.

## Приложения-клиенты, использующие ODS

Следующие приложения-клиенты IndustrialSQL Server до сих пор используют ODS-сервер IndustrialSQL для доступа к данным: InSQL Trend, InSQL QuickLook, InSQL VectorMaster и VectorViewer, Industrial Workbook, IndustrialSQL ActiveTrend и элементы управления ActiveEvent ActiveX. Эти приложения-клиенты работают по принципу "интеллектуального" соединения, устанавливая прозрачное соединение с сервером базы данных IndustrialSQL, не взирая на версию системы IndustrialSQL Server, с которой устанавливается соединение.

Подробнее см. в главе 12, "Поддержка клиентов".

## Использование расширений временного домена IndustrialSQL через ODS

Для выборки данных через ODS расширения временного домена IndustrialSQL Server (например, расширения счетчика строк и разрешения) работают как хранимые процедуры и выражения SET.

Подробнее см. в главе 14, "Выборка данных", *Справочного руководства IndustrialSQL Server*.

## Выборка через поставщика данных OLE DB

OLE DB (Object Linking and Embedding for Databases) — это интерфейс программирования приложений (API), позволяющий приложениям-клиентам на COM-основе получать доступ к данным, не хранящимся физически в базе Microsoft SQL Server, к которой они присоединяются.

Преимущество использования OLE DB заключается в том, что этот метод обеспечивает более широкий доступ к различным типам данных, чем это было раньше. Используя OLE DB, можно одновременно получать доступ к данным из различных источников, таких как базы данных Microsoft SQL Server, Oracle и Microsoft Access. Запрос, обращающийся к данным из различных непохожих источников, называется "разнородным запросом", где "разнородный" означает "непохожий". Разнородные запросы также называются "распределенными", так как данные могут быть распределены по различным источникам данных.

Microsoft SQL Server использует OLE DB для обработки разнородных запросов и для упрощения связывания данных между различными источниками. Благодаря OLE DB, Microsoft SQL Server поддерживает запросы Transact-SQL на данные, хранимые в одном или нескольких серверах SQL и разнородных базах данных, без необходимости использования специальных шлюзовых серверов, таких как сервер базы данных IndustrialSQL.

Следующие разделы описывают концепции OLE DB более детально.

### Нелокальные хранилища данных

Нелокальными источниками данных называются просто хранилища данных, существующие вне базы данных Microsoft SQL Server. Microsoft определяет эти типы как "хранилища данных". В случае с IndustrialSQL Server, нелокальное хранилище данных — это набор файлов архивных блоков, которые содержат аналоговые, дискретные и архивные данные. OLE DB используется для доступа к данным, находящимся в любом нелокальном хранилище. Этот доступ осуществляется через такой программный компонент, как поставщик данных OLE DB.

### Поставщики данных OLE DB

Интерфейс, требуемый для обращения к данным архивных блоков IndustrialSQL Server (хранилище данных OLE DB), обеспечивается "виртуальным" севером, или поставщиком данных OLE DB. Поставщик данных OLE DB позволяет использовать мощность обработчика запросов Microsoft SQL Server для упрощения и ускорения выборки данных из базы Microsoft SQL Server и архивных блоков. Кроме того, поставщик данных OLE DB IndustrialSQL Server обладает более широкими возможностями запросов для служб открытых данных (ODS).

Поставщик данных OLE DB для IndustrialSQL Server называется "INSQL". Поставщик данных OLE DB INSQL устанавливается одновременно с IndustrialSQL Server и затем соединяется с системой Microsoft SQL Server.

Подробнее о синтаксисе поставщика данных OLE DB INSQL см. в разделе "Связь поставщика данных OLE DB INSQL с Microsoft SQL Server".

Поставщик данных OLE DB INSQL должен находиться на том сервере, где запущен Microsoft SQL Server. Набор операций Transact-SQL, используемый для выборки данных из архивных блоков, зависит от возможностей поставщика данных OLE DB INSQL. Поставщик данных OLE DB INSQL является приложением, совместимым с SQL-92.

Подробнее см. в главе 14, "Выборка данных" в *Справочном руководстве IndustrialSQL Server*.

## Удаленные таблицы OLE DB

Как сервер базы данных ODS IndustrialSQL, так и поставщик данных OLE DB IndustrialSQL отображают данные архивных блоков как обычные таблицы сервера SQL. В случае с сервером базы данных ODS, эти таблицы называются "таблицами расширения" (термин Wonderware). В случае с OLE DB, они называются "удаленными таблицами" (термин Microsoft).

Подробнее об удаленных таблицах IndustrialSQL Server см. в главе 11, "Удаленные таблицы OLE DB" в *Справочном руководстве IndustrialSQL Server*.

## Архитектура поставщика данных OLE DB INSQL

Для доступа к архивным данным IndustrialSQL Server через OLE DB любой COM-совместимый клиент, такой как InSQL Report, должен напрямую соединиться с Microsoft SQL Server и затем указать в синтаксисе запроса имя поставщика данных OLE DB INSQL. Это полностью отличается от того способа, которым запросы обрабатывались в версиях IndustrialSQL Server до 7.1, где приложения-клиенты должны были устанавливать соединение с сервером базы данных IndustrialSQL ODS.

Подробнее см. в разделе "Архитектура ODS IndustrialSQL Server" ранее в этой главе.

При подаче запроса, в котором указано имя поставщика данных OLE DB INSQL, система Microsoft SQL Server передаст нужные части запроса поставщику данных OLE DB сервера INSQL. Поставщик данных OLE DB INSQL затем контактирует с системой сбора IndustrialSQL Server для обнаружения хранилища данных, распаковки и выдачи запрошенной в Microsoft SQL Server в виде набора строк. Microsoft SQL Server производит все остальные процедуры и выдает данные приложению-клиенту как набор результатов и, если возможно, вместе с набором параметров вывода.

Подробнее об этой архитектуре см. в разделе "Сводка ODS и OLE DB" далее в этой главе.

Подробнее о типах запросов см. в главе 14, "Выборка данных" в *Справочном руководстве IndustrialSQL*.

## Подготовка к использованию OLE DB

Для выполнения разнородных запросов должны быть выполнены следующие условия:

- Используемая база данных *не* должна запускаться в режиме совместимости с версией 6.5.
- При соединении с базой данных должны быть включены переменные ANSI - nulls и warnings.

Правильные настройки этих переменных являются стандартом для связи ODBC и OLE DB (или ADO). Клиенты, не нуждающиеся в переменных ANSI:

- SQL Server Query Analyzer
- MS Query - программа, входящая в состав Microsoft Office.
- Любое приложение, использующее ODBC, ADO или OLE DB для связи с SQL Server.

Правильные настройки переменных ANSI не являются стандартом для связи с DB-Library. Клиенты, которые должны использовать переменные ANSI:

- ISQL/w - программа из SQL Server 6.5, поскольку это клиент DB-Library.
- Клиенты, соединяющиеся с сервером базы данных InSQL (PDSSRV.EXE), так как он постоянно использует связь с SQL Server через DB-Library.
- Любые другие приложения, использующие DB-Library для связи с SQL Server.

ANSI-переменные nulls и warnings можно включить, используя следующие выражения:

```
SET ANSI_NULLS ON  
  
SET ANSI_WARNINGS ON  
  
go
```

## Клиенты IndustrialSQL, использующие OLE DB

Следующие приложения-клиенты IndustrialSQL Server используют OLE DB для доступа к данным: InSQL Configure, InSQL Report и элементы ActiveX IndustrialSQL: ActiveDataGrid, ActiveGraph, ActiveTimeSelector и ActiveTagBrowser.

Подробнее см. в главе 12, "Связь с клиентом".

## Использование расширений временного домена IndustrialSQL через OLE DB

Для выборки данных через OLE DB расширения временного домена IndustrialSQL Server (например, счетчик строк и разрешение) включаются в выражения SQL как часть оператора WHERE.

Подробнее см. в главе 14, "Выборка данных" в *Справочном руководстве IndustrialSQL Server*.

## Связь поставщика данных OLE DB INSQ L с Microsoft SQL Server

Поскольку поставщик данных OLE DB INSQ L получает данные из архивных блоков и "передает" их в Microsoft SQL Server в виде набора строк, можно считать его разновидностью сервера. Поставщик данных OLE DB INSQ L необходимо зарегистрировать в системе Microsoft SQL Server как "связанный" сервер прежде, чем его можно будет использовать для обработки запросов.

Эта связь происходит автоматически во время установки IndustrialSQL Server. Если для какой-либо цели потребуется повторно связать поставщика данных OLE DB INSQ L с Microsoft SQL Server, выражения для этого выглядят следующим образом:

```
sp_addlinkedserver
  @server = 'INSQ L',
  @srvproduct = '',
  @provider = 'INSQ L'
go
```

```
sp_addlinkedsrvlogin 'INSQ L', 'FALSE', NULL, NULL, NULL
go
```

Заметьте, что "INSQ L" — это имя поставщика данных OLE DB INSQ L, определенного как связанного сервера. Используйте это имя в запросе для указания поставщика данных OLE DB INSQ L.

Для связки между таблицами аналоговых и дискретных архивов программа установки создает псевдоним того же поставщика данных OLE DB INSQ L:

```
sp_addlinkedserver
  @server = 'INSQ LD',
  @srvproduct = '',
  @provider = 'INSQ L'
go
```

```
sp_addlinkedsrvlogin 'INSQ LD', 'FALSE', NULL, NULL, NULL
go
```

Например, если требуется обработать запрос, выполняющий этот тип связки, используйте обычный псевдоним при указании первой таблицы (таблица аналоговых архивов), и второй псевдоним — при указании второй таблицы (таблица дискретных архивов, поэтому к имени псевдонима добавляется "D").

## Сводка ODS и OLE DB

В следующей таблице сведены различия между использованием ODS и OLE DB как части IndustrialSQL Server для выборки данных.

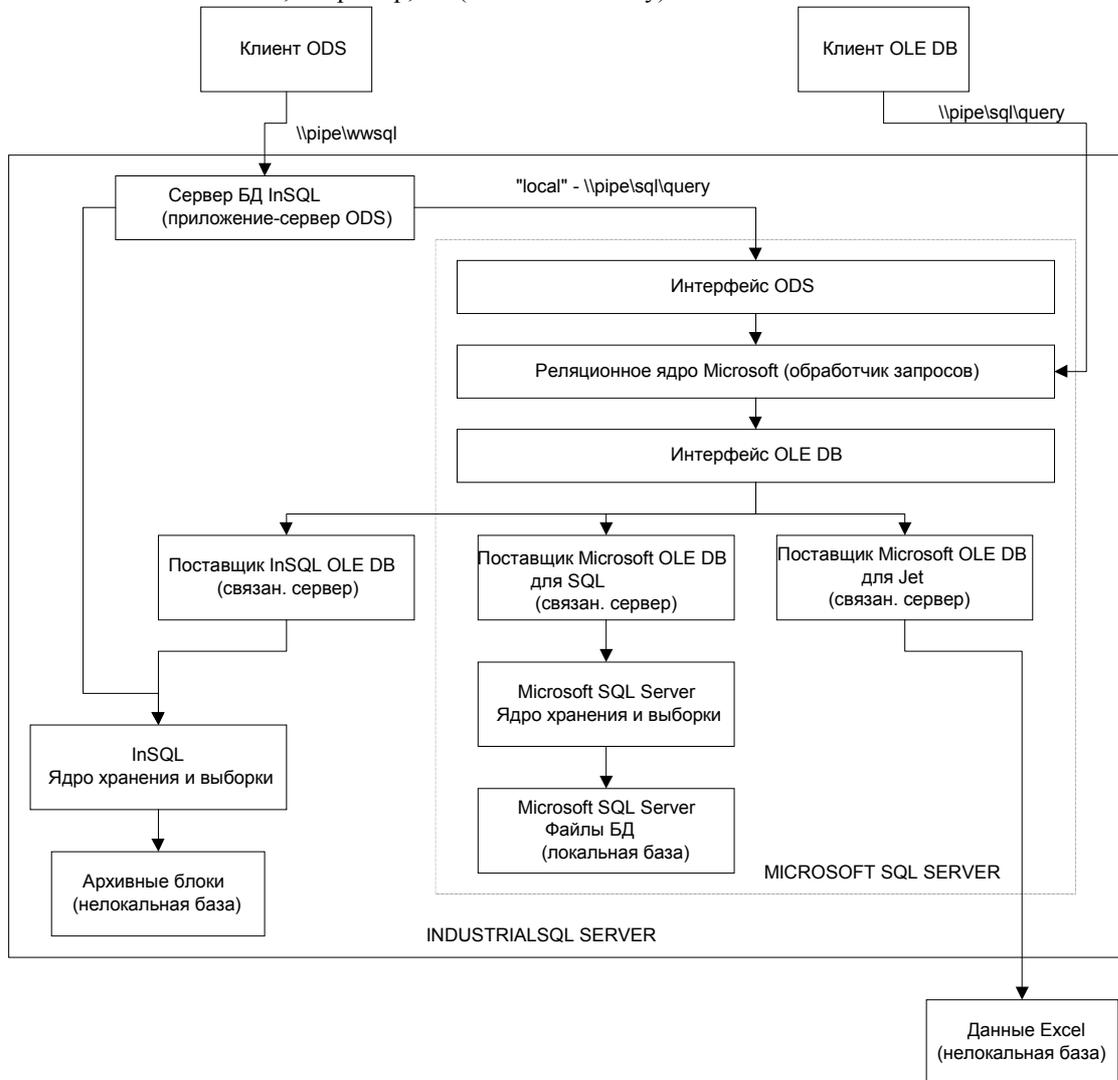
	ODS	OLE DB
Назначение:	Может обращаться к данным, хранимым в файлах (архивных блоках) вне Microsoft SQL Server.	Может обращаться к данным, хранимым в файлах (архивных блоках) вне Microsoft SQL Server.
Поддержка Transact-SQL:	Ограниченная поддержка Transact-SQL. Не может производить связки таблиц хранения данных и любых других таблиц базы данных. Не может проводить связки между множественными непохожими хранилищами данных.	Мощная поддержка Transact-SQL. Может проводить связки между множественными непохожими хранилищами данных.
Стандартное соединение с базой данных:	Соединение IndustrialSQL Server: \\имякомпьютера\pipe\sql\query	Соединение Microsoft SQL Server: \\имякомпьютера\pipe\sql\query
Альтернативное соединение с базой данных:	Соединение Microsoft SQL Server: \\имякомпьютера\pipe\local\query	Соединение IndustrialSQL Server: \\имякомпьютера\pipe\wwsql
Стандартное используемое имя базы данных:	InSQL	Имя Microsoft SQL Server (обычно, имя компьютера).
Разбор запросов клиента:	Запросы клиента разбираются сервером базы данных IndustrialSQL Server. Сервер базы данных посылает все запросы к архивным данным в систему сбора IndustrialSQL и пропускает запросы выборки данных из базы данных SQL в Microsoft SQL Server.	Все запросы клиента разбираются обработчиком запросов Microsoft SQL Server. Обработчик запросов отсылает все архивные данные поставщику данных OLE DB InSQL, который затем передает их механизму сбора IndustrialSQL.
Поддержка расширений временного домена IndustrialSQL (Industrial-SQL):	Использование выражений SET и хранимых процедур. Например, SET RowCount, ww_RowCount.	Настройка расширений временного домена выполняется с помощью оператора WHERE.
Данные, хранящиеся в архивном блоке выглядят так:	Таблицы расширения (термин Wonderware), включающие в себя столбцы <i>DateTime</i> , <i>TagName</i> , <i>Value</i> , <i>Quality</i> и <i>QualityDetail</i> .	Удаленные таблицы (термин Microsoft) включающие в себя столбцы <i>DateTime</i> , <i>TagName</i> , <i>Value</i> , <i>Quality</i> и <i>QualityDetail</i> , плюс допустимые виртуальные столбцы параметров (например, <i>wwRowCount</i> ).

(продолжение следует)

*(продолжение)*

Идентификация архивной таблицы в запросе:	Используется имя таблицы расширения: Select * from AnalogHistory	Используется обозначение из четырех частей для идентификации удаленной таблицы: Select * from InSQL.Runtime.Dbo.AnalogHistory. Или используется имя соответствующего представления: Select * from v_AnalogHistory
Приложения-клиенты, использующие технологию:	InSQL Trend InSQL QuickLook InSQL VectorMaster и VectorViewer Industrial Workbook ActiveTrend и ActiveEvent	InSQL Configure InSQL Report ActiveDataGrid, ActiveGraph, ActiveTimeSelector и ActiveTagBrowser

Следующая диаграмма показывает схему доступа клиентов ODS или OLE DB к данным. Клиент ODS обращается к данным в архивных блоках и данным конфигурации в базе данных *Runtime*. Клиент OLE DB обращается к архивным данным в архивных блоках, данным настройки в базе данных конфигурации и к данным любого связанного поставщика данных OLE DB как, например, Jet (включен в схему).



# Система управления событиями

В этой главе описана система управления событиями. Здесь рассмотрены все аспекты работы этой системы.

## Содержание

- О системе управления событиями
- Определение событий: тэги событий
- Детекторы событий
- Действия над событием
- Выделение системного потока для событий
- Определение края
- Архивная запись событий
- Компоненты и программы управления событиями
- Таблицы хранения событий
- Таблицы сводок
- Настройка событий
- Выборка данных зарегистрированных событий

## О системе управления событиями

Система управления событиями позволяет администраторам настроить механизм распознавания событий и выполнения соответствующих ответных действий. Проще говоря, когда происходит что-то заслуживающее внимания, это называется *событием*. Систему управления событиями можно настроить на периодическую проверку возникновения событий. Это называется распознаванием (фиксацией) события, после которого может быть активизировано *действие*.

---

**Примечание.** Система управления событиями не является системой реального времени; наоборот, она работает с архивными данными. Для обработки событий в реальном времени используются приложения типа InTouch.

---

В IndustrialSQL Server при сохранении события сохраняется больше информации, чем только сам факт события. Определение события представляет собой набор атрибутов, описывающих момент, когда значения архивных тэгов IndustrialSQL Server удовлетворили набору критериев. Атрибуты включают в себя дату и время возникновения события, дату и время его фиксации и критерии, которым оно удовлетворяет.

Система управления событиями выполняет следующие основные функции:

- Используя заранее определенные критерии, фиксирует момент возникновения события.
- Выборочно регистрирует возникновение событий в архиве.
- Включает настроенное событие всегда, когда условия удовлетворяют требуемым критериям.

---

**Примечание.** Система управления событиями не сбивается при потере связи с IndustrialSQL Server и возвращается к работе при восстановлении связи.

---

## Определение событий: тэги событий

*Тэгом события* называется определение события в системе. Например, если требуется регистрировать событие, когда температура танкера достигла 100 градусов, следует определить тэг события и назвать его "TankAt100". Тэги событий отличаются от всех других типов тэгов IndustrialSQL Server (аналоговые, дискретные и текстовые). Эти типы тэгов являются определениями переменных, требующих сохранения, но тэг события — это имеющая имя ссылка для определения конкретного события, возникновение которого требуется фиксировать и затем применять какое-либо действие.

Тэги событий создаются и управляются программой InSQL Configure. При создании тэга события следует указать:

- Имя, описание и другие основные настройки.
- Критерии события, описывающие условия для фиксации события и интервал, с которым система управления событиями должна проверять, не произошло ли событие.
- Следует ли записывать момент возникновения события.
- Следует ли включать/выключать фиксацию события.
- Действие, которое будет предпринято при фиксации события

В следующей таблице приведены таблицы базы данных, используемые для хранения определений тэгов событий:

Тип определения	Таблица
Выявление конкретного события и возможное ответное действие	<i>EventTag</i> , <i>DetectorType</i> и <i>ActionType</i>
Обнаружение события, основанного на времени	<i>TimeDetectorDetail</i>
Снимки аналоговых, дискретных и текстовых тэгов	<i>SnapshotTag</i>
Операции сводки	<i>SummaryOperation</i> и <i>SummaryTagList</i>

## Детекторы событий

Каждый тэг события имеет соответствующий детектор события. Детектор событий — это механизм для определения момента, когда все критерии тэга события удовлетворены. Тип детектора событий должен быть указан в таблице *DetectorType*. При настройке детектора событий следует сначала указать его тип, а затем параметры, соответствующие этому типу детектора.

В IndustrialSQL Server изначально присутствуют заранее настроенные детекторы событий.

Детектор	Описание
Внешний	<p>Состояние события включается элементом ActiveEvent.</p> <p>Например, сценарий InTouch может активировать необходимые методы ActiveEvent для включения события IndustrialSQL. Внешний детектор не настраивается.</p>
Общий SQL	<p>Выявляет событие на основе критериев, описанных в выражении SQL.</p> <p>Можно использовать уже настроенные шаблоны SQL, хранящиеся в базе данных, как основу для сценария или же можно создать собственный сценарий "с нуля".</p>
Указанное аналоговое значение	<p>Выявляет событие на основе критериев, применимых к значению конкретного аналогового тэга.</p> <p>Значение аналогового тэга связывается (с помощью операторов) с постоянным значением.</p>
Указанное дискретное значение	<p>Выявляет событие на основе критериев, связанных с состоянием конкретного дискретного тэга.</p> <p>Состояние дискретного тэга сравнивается с постоянным целевым значением.</p>
Планировщик	<p>Обнаруживает событие на основе времени системных часов. Например, можно записывать информацию о событии каждый понедельник в 2:00 p.m.</p>

Для детекторов на основе SQL, в зависимости от настройки события, время фиксации события может наступить сразу же после фактического времени его возникновения. Например, детектор настроен на фиксацию конкретного события, лежащего в промежутке времени в 10000 мс (10 секунд). Это означает, что каждые 10 секунд детектор будет проверять, произошло ли событие. Если событие происходит через 2 секунды (2000 мс) после последней проверки, детектор событий не засечет это событие, пока не истекнут положенные 10 секунд.

Детекторы-планировщики однако являются детекторами реального времени; они запускаются каждую 1 секунду.

## Действия над событием

Событие может быть связано или не связано с *действием*. Действие над событием настраивается для совершения в момент фиксации события детектором. Действия над событиями необязательны; бывают случаи, когда нужно только зафиксировать факт возникновения события.

В комплект IndustrialSQL Server входят следующие заранее настроенные действия над событиями.

Детектор	Описание
None	Не производит никаких действий при фиксации события. Не производит никаких действий при выявлении события.
Generic SQL	Производит действие, указанное в выражении SQL Можно использовать заранее настроенные шаблоны SQL, хранящиеся в базе данных, как основу для сценария, или можно создать собственный сценарий полностью "с нуля".
Snapshot	Записывает значения выбранного аналогового тэга или состояния выбранных дискретных тэгов, или текстовые значения выбранных текстовых тэгов в то время, когда произошло событие. Также записывается качество.
E-mail	Посылает заранее настроенное электронное письмо Microsoft Exchange.
Deadband	Изменяет мертвую зону времени и/или значения хранения для одного или нескольких аналоговых тэгов, использующих дельта-хранение (хранение методом исключения).
Summary	Проводит одну или несколько агрегаций для одного или нескольких аналоговых тэгов.

## Выделение системного потока для событий

Системный поток - это объект, который выполняет независимую функцию в определенном процессе. В системе управления событиями потоки выделяются для детекторов событий и действий.

### Выделение потока для детектора

Для фиксации событий выделяются два потока: один для детекторов на основе SQL и один для детекторов-планировщиков. Каждый из этих потоков использует учетную запись на сервере SQL.

Детекторы-планировщики работают в реальном времени и выполняются в запланированное время. Так как детекторы на основе SQL работают с архивными данными, они сначала упорядочиваются, а затем выполняются. Сначала детекторы делятся на группы на основе временного интервала, указанного для тэгов событий. Временной интервал — это частота, с которой настроенный компонент детектора проверяет, произошло ли событие. (Этот временной интервал хранится в столбце *ScanRate* таблицы *EventTag*). Затем детекторы последовательно упорядочиваются в группе временных интервалов.

---

**Примечание.** Каждый поток детектора поддерживает соединение с базой данных.

---

### Задержка

Задержка — это временной разрыв между фактическим возникновением события и его фиксацией системой. Задержку важно учитывать, когда требуется быстро включить действие после возникновения события.

Например, если создано событие, которое фиксирует момент, когда измерительный прибор начинает выдавать ненормальные значения, то может возникнуть желание настроить действие для оповещения о необходимости скорейшей проверки этого прибора. Следует создать тэг события с коротким интервалом (частотой сканирования), чтобы система управления событиями достаточно часто проверяла, произошло ли событие.

Иные события (например, когда оператор закончил свою смену) не столь критичны, и для них фактор задержки не столь важен.

Соблюдайте осторожность при присвоении временного интервала тэгам событий. Слишком много тэгов событий с одинаковым интервалом времени могут вызвать "переполнение" детектора. Хотя это и не окажет негативного влияния на фиксацию событий, это может привести к увеличению задержки. Кроме того, небольшие интервалы времени могут привести к чрезмерной загрузке процессора, что приведет к замедлению системы.

## Выделение потока для действия

Все действия привязаны к пулу потоков. В этом пуле имеется три очереди:

- "Критическая" очередь, в которой содержатся любые действия для тэгов событий, которые получили критический приоритет. Любые, особенно важные, события можно поставить в категорию "критических" событий. Если происходит перегрузка системы, все события, помеченные "критическим приоритетом" будут обрабатываться в первую очередь. Будьте внимательны, присваивая событию метку "критическое". Если все события помечены как критические, то в случае перегрузки в системе отсутствует защита. Метка "критическое" означает, что система будет всегда стараться выполнить данное действие перед всеми обычными действиями.
- "Нормальная" очередь, содержащая любые действия для тэгов событий, которые получили нормальный приоритет. Если происходит перегрузка системы, все некритические события, помеченные как "нормальные", обрабатываются после критических.
- Задержанная очередь, в которой содержатся все действия для тэгов событий, которым присвоена задержка детектора. Задержка детектора — это минимальное количество времени, которое должно пройти после фиксации события до исполнения соответствующего действия.

---

**Примечание.** Каждый поток в пуле потоков поддерживает связь с базой данных.

---

## Определение края

При фиксации событий полезным является сужение количества строк в наборе результатов, где удовлетворение критериям изменялось с истины на ложь и наоборот. Например, может понадобиться узнать, когда уровень танкера понизится до 5 футов. Пока уровень танкера не превышает 5 футов, критерии на являются истиной. Только когда уровень пересекает абстрактную линию между неудовлетворением и удовлетворением критерия, событие действительно происходит. Абстрактная "линия", на которой происходит изменение называется *краем*.

За какой-то период времени может неоднократно случиться так, что критерии пересекают край от "удовлетворенных" на "неудовлетворенные" и наоборот. Значения с любой стороны этого края могут быть зафиксированы, если тэг события настроен для добавления этой информации. Существует четыре параметра для определения края: *нет*, *ведущий*, *ведомый* или *оба*. При использовании этих параметров выдаются различные результаты:

Определения края	Результаты
None	Выдает все строки, удовлетворяющие критериям. При отсутствии определения края действует указанное разрешение.
Leading	Выдает только те строки, которые первыми удовлетворили критериям (выдана истина) после строки, не удовлетворяющей им (выдана ложь).
Trailing	Выдает только строки, которые первыми не удовлетворили критериям (выдана ложь) после строки, удовлетворяющей им (выдана истина).
Both	Выдаются все строки, удовлетворяющие и ведущим, и ведомым параметрам.

Определение края действует только для детекторов аналоговых и дискретных значений. Кроме того, определение края действует по-разному, в зависимости от того, используются ли аналоговые или дискретные тэги.

Подробнее см. в главе 14, "Выборка данных", в *Справочном руководстве IndustrialSQL Server*.

## Архивная запись событий

Точное время, когда действительно произошло конкретное событие и время, когда оно было зафиксировано системой управления событиями, могут быть записаны в таблицу IndustrialSQL Server, называемую *EventHistory*. Каждая строка в таблице *EventHistory* содержит имя определенного тэга события, метку даты/времени удовлетворения критериям фиксации для этого тэга события, метку даты/времени фиксации события детектором и дополнительное поле, предоставляющее информацию о фиксации события.

---

**Примечание.** Запись событий нельзя отключить, если настроен аналоговый и/или дискретный снимок.

---

Помимо условий, которые должны быть выполнены для возникновения события, должен также быть включен процесс фиксации событий для тэга события.

Подробнее о включении/выключении события см. в разделе "Редактирование общей информации тэга события" далее в этой главе.

---

**Совет.** Таблицу *EventHistory* можно использовать для определения того, когда слишком много тэгов имеют один и тот же временной интервал. Если задержка между моментом, когда событие действительно произошло (хранится в столбце *DateTime*), и моментом, когда оно было зафиксировано (столбец *DetectDateTime*), постоянно растет и/или фиксируется множество событий, происходящих в один и тот же интервал, то следует переместить некоторые детекторы событий на другой временной интервал.

Подробнее см. в разделе "Выделение системного потока для событий" ранее в этой главе.

---

## Компоненты и программы управления событиями

Система управления событиями состоит из следующих компонентов и программ:

- Модель данных системы управления событиями (Event System Data Model), определяющая схему конфигурации и архивной записи данных о событиях.
- Служба системы управления событиями (ESS), которая является службой Windows NT, координирующей функции фиксации событий и выполнения действий. ESS автоматически запускается и останавливается при запуске и остановке системы IndustrialSQL Server через программу InSQL Control; вместе с тем, ESS может запускаться и останавливаться независимо от остальной системы через программу Службы в Панели управления.
- Набор системных тэгов событий для внутренних системных переменных.

- InSQL Configure, программа настройки системы, позволяющая создавать или удалять тэги событий, настраивать детекторы событий и действия над событиями.
- COM-компоненты системы управления событиями для нумерации, детекторов и действий.

## Переменные системы управления событиями

Подсистема управления событиями использует набор переменных для облегчения фиксации событий и действий. Цель этих переменных — в облегчении создания запросов пользователем (или редактором конфигурации). Компоненты системы заменяют эти переменные соответствующими значениями перед непосредственным выполнением запроса. Запрос, получаемый системой IndustrialSQL Server, никогда не содержит переменных.

Переменные выглядят следующим образом:

Переменная	Описание
<i>@EventTime</i>	Время фиксации события текущим детектором.
<i>@EventTagName</i>	Имя тэга, связанное с зафиксированным событием.
<i>@StartTime</i>	Начальное время выполнения запроса детектора.
<i>@EndTime</i>	Начальное время выполнения запроса детектора.

Обычно, запрос фиксации, выполняемый компонентом детектора, будет таким:

```
SELECT DateTime
FROM AnalogHistory
WHERE Tagname = 'BoilerPressure' AND Value > 75
AND DateTime > '@StartTime'
AND DateTime < '@EndTime'
```

*@StartTime* и *@EndTime* являются просто заглушками, позволяющими детектору координировать фиксацию событий при изменении временного диапазона.

Следующий запрос действия показывает, как могут использоваться переменные событий:

```
SELECT * INTO TEMPTABLE FROM AnalogHistory
WHERE DateTime = '@EventTime'
AND TagName IN
(SELECT TagName
FROM SnapshotTag
WHERE EventTagName = '@EventTagName'
AND TagType = 1)
```

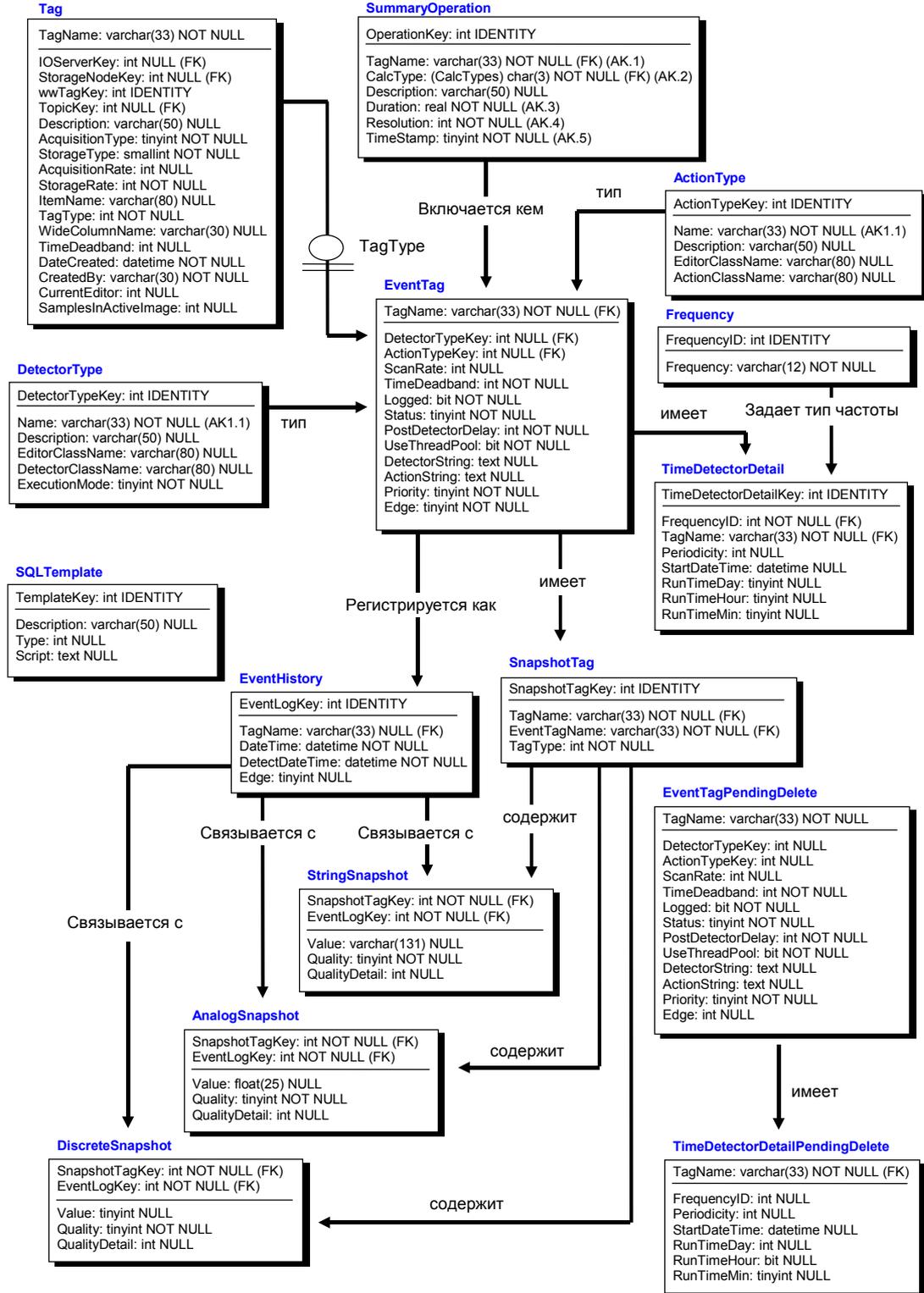
---

**Примечание.** Эти переменные работают только во внутреннем контексте системы управления событиями и не подходят к запросам из инструментов клиента, таких как SQL Server Query Analyzer и т.п.

---

# Таблицы хранения событий

Таблицы хранения событий показаны ниже.



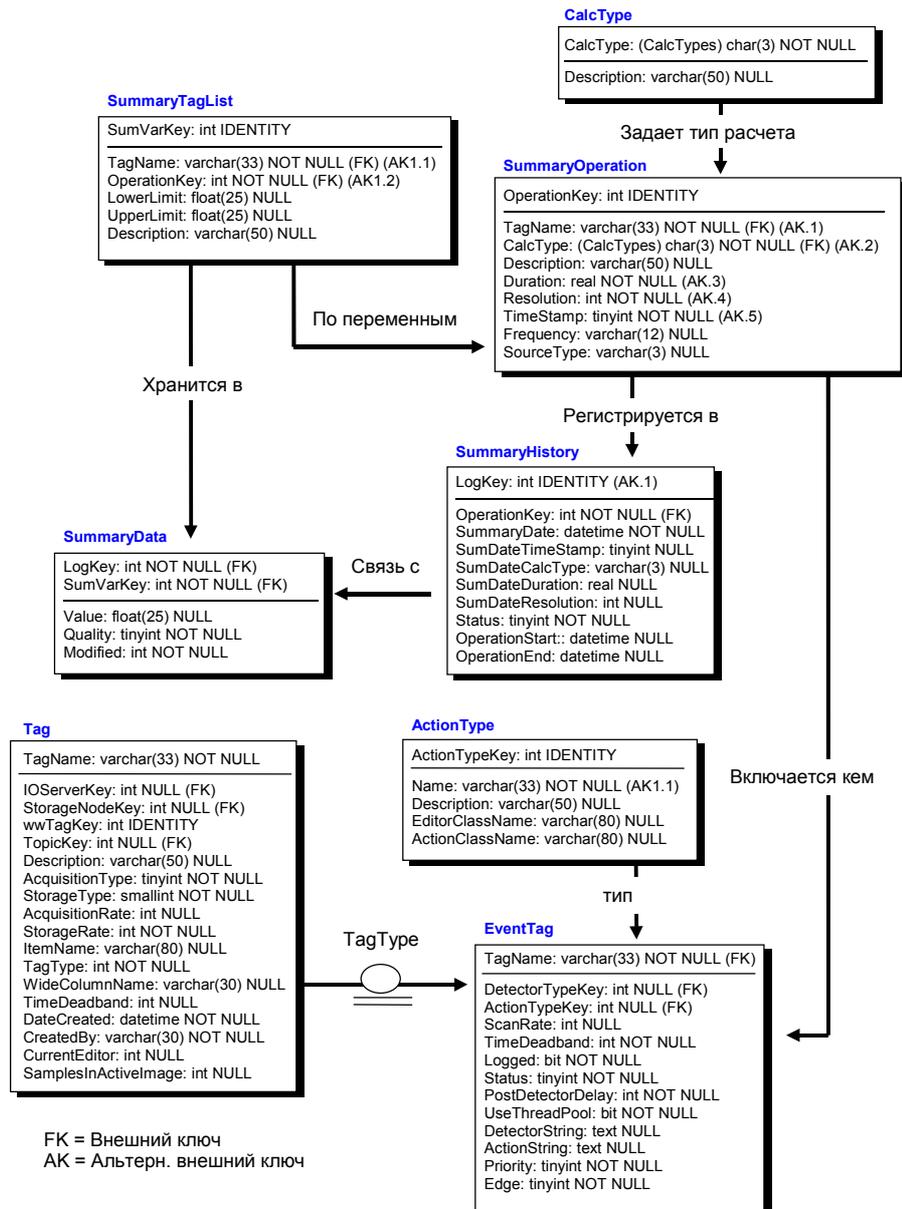
FK = Внешний ключ  
AK = Альтерн. внешний ключ

# Таблицы сводок

Действия со сводками есть определенные расчеты, проводимые для одного или нескольких аналоговых тэгов. Для выполнения действия со сводкой система управления событиями должна быть обеспечена одним или несколькими тэгами.

Архивные результаты действий со сводками хранятся в таблицах *SummaryHistory* и *SummaryData*.

Схема таблицы сводок выглядит следующим образом:



## Настройка событий

События легко настраиваются при помощи программы InSQL Configure. Все определенные тэги событий программа InSQL Configure отображает в папке браузера **Event Tags**. Для просмотра или изменения тэга события нужно просто выбрать тэг в браузере. Информация о тэге появится в панели свойств.

Можно настроить общие детали, детали действий и детали детекторов для выбранного тэга события, а также можно добавить в систему новые тэги событий.

➤ **Как запустить InSQL Configure и просмотреть тэги событий:**

1. Из меню **Start** на панели задач Windows выберите программную группу Wonderware FactorySuite, затем подгруппу IndustrialSQL Server, а в ней выберите InSQL Configure.
2. Введите данные учетной записи для доступа к IndustrialSQL Server.
3. Появится главное окно программы InSQL Configure.
4. В браузере разверните папку **Public Groups**.
5. Выберите папку **All Event Tags**. Все настроенные тэги событий появятся в нижней панели браузера.

[Подробнее см. в главе 2, "InSQL Configure"](#).

## Редактирование общей информации тэга события

Общая информация тэга события включает в себя только информацию, относящуюся к определению тэга. Детекторы событий и действия определяются по-отдельности и затем присваиваются тэгу события.

➤ **Как изменить общую информацию тэга события:**

1. Выберите в браузере тэг события.
2. Выберите вкладку свойств **General**.

The image shows a configuration window with three tabs: 'General', 'Detector', and 'Action'. The 'Detector' tab is selected. It contains the following fields and controls:

- Description:** A text box containing 'Status Tag snapshot event'.
- Enabled:** A checked checkbox.
- Logged:** A checked checkbox.
- Detect on the edge...:** A dropdown menu with 'LEADING' selected.
- Post Detector Delay:** A text box with '0' and '(ms)' next to it.
- Priority:** A dropdown menu with 'NORMAL' selected.

3. В поле **Description** введите описание тэга события.
4. Установите флажок **Enabled** для активации тэга события.
5. Установите флажок **Logged** для регистрации событий этого тэга в таблице *EventHistory*.

---

**Примечание.** Регистрацию событий нельзя выключить, если настроены аналоговые, дискретные или текстовые снимки.

---

6. Для изменения определения края выберите тип определения из списка **Detect on the edge**.

---

**Примечание.** Определение края не действует для внешних детекторов или планировщиков.

---

#### **None**

Выдает все строки, успешно удовлетворившие критерию; определения края не происходит.

#### **Leading**

Выдает только те строки, которые первыми удовлетворили критериям (выдана истина) после строки, не удовлетворившей им (выдана ложь).

#### **Trailing**

Выдает только строки, которые первыми не удовлетворили критериям (выдана ложь) после строки, удовлетворившей им (выдана истина).

#### **Both**

Выдаются все строки, удовлетворяющие обоим параметрам определения края.

Подробнее см. в разделе "Определение края" ранее в этой главе.

7. Чтобы установить задержку выполнения любых связанных с детектором действий, введите значение в поле **Post Detector Delay**.

Задержка детектора есть количество времени (в мс), которое должно пройти после фиксации события до выполнения соответствующего действия.

8. Выберите приоритет из списка **Priority**.

**Critical**

Используется только для особенно важных событий. Если происходит перегрузка системы, события с критическим приоритетом всегда обрабатываются первыми. Например, критическим событием может быть момент, когда температура достигает опасного уровня, после чего вам посылается электронное сообщение.

**Normal**

Используется для нормальных событий. События с нормальным приоритетом обрабатываются после всех критических событий. Примером нормального события является ежедневная сводка.

9. Для сохранения результатов нажмите кнопку . Или же в меню **File** выберите **Save**.
10. Для отмены изменений нажмите кнопку . Или же в меню **File** выберите **Undo**.

## Создание тэга события

*Тэгом события* называется определение события в системе. При создании тэга события в таблице *EventTag* появляется новая строка.

### ➤ Как создать тэг события:

1. В меню **File** выберите **New**. Или нажмите кнопку  на панели инструментов.

Появится диалоговое окно **New**.

**Ошибка! Указано неверное имя файла.**

---

**Совет.** Нажмите в этом окне правой кнопкой мыши для выбора режима отображения ярлыков.

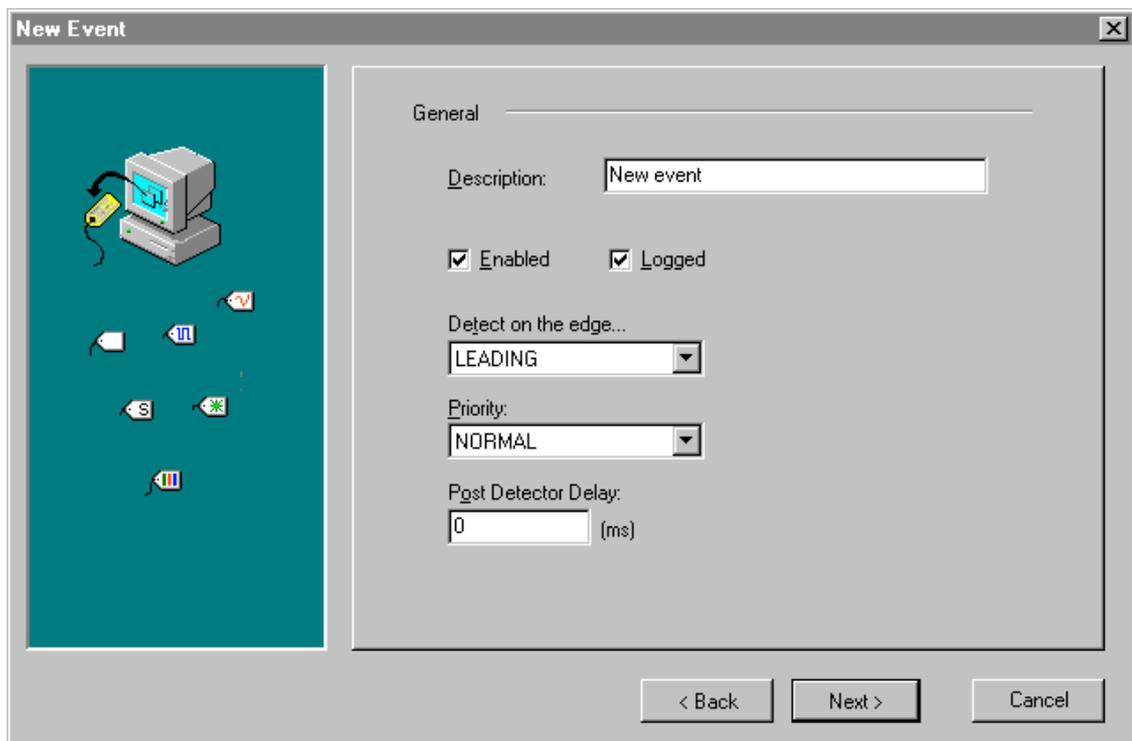
---

2. Выберите ярлык **Event** и нажмите **OK**.

Появится диалоговое окно **New Event**.

3. Введите уникальное имя тэга события и нажмите **Next**. В имени тэга нельзя использовать пробелы, кавычки и другие специальные символы, хотя InSQL Configure позволяет их вводить.

Затем мастер **New Event** просит ввести общую информацию о тэге события.

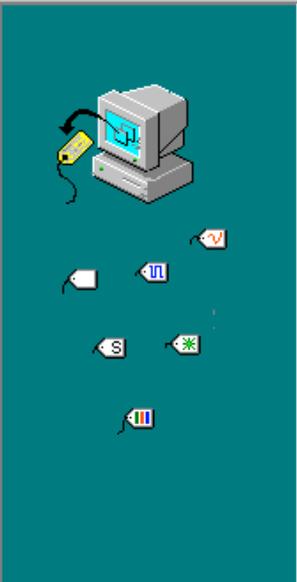


4. Настройте общие параметры тэга события и нажмите **Next**.

Подробнее см. в разделе "Редактирование общей информации тэга события" ранее в этой главе.

Затем мастер **New Event** просит настроить детектор для тэга события.

**New Event** [X]



Time Interval:  (ms)

Detector Type:

Tag Name:  Operator:  Constant Value:

Data Resolution:  (ms)

Note: Data Resolution is only applicable when the General tab's "Detection on the edge..." dropdown is set to NONE.

< Back    Next >    Cancel

5. Настройте детектор для тэга события. Детектор может быть внешним, общим SQL, указанным аналоговым значением, указанным дискретным значением и планировщиком. Нижняя часть панели свойств изменится в зависимости от выбранного типа детектора.

**Time Interval**

Интервал в миллисекундах, за который обрабатываются условия события. Если условия события удовлетворены, тогда событие сохраняется и все связанные с ним действия включаются. Временной интервал должен быть более 500 миллисекунд. Временной интервал используется только с детекторами на основе SQL.

---

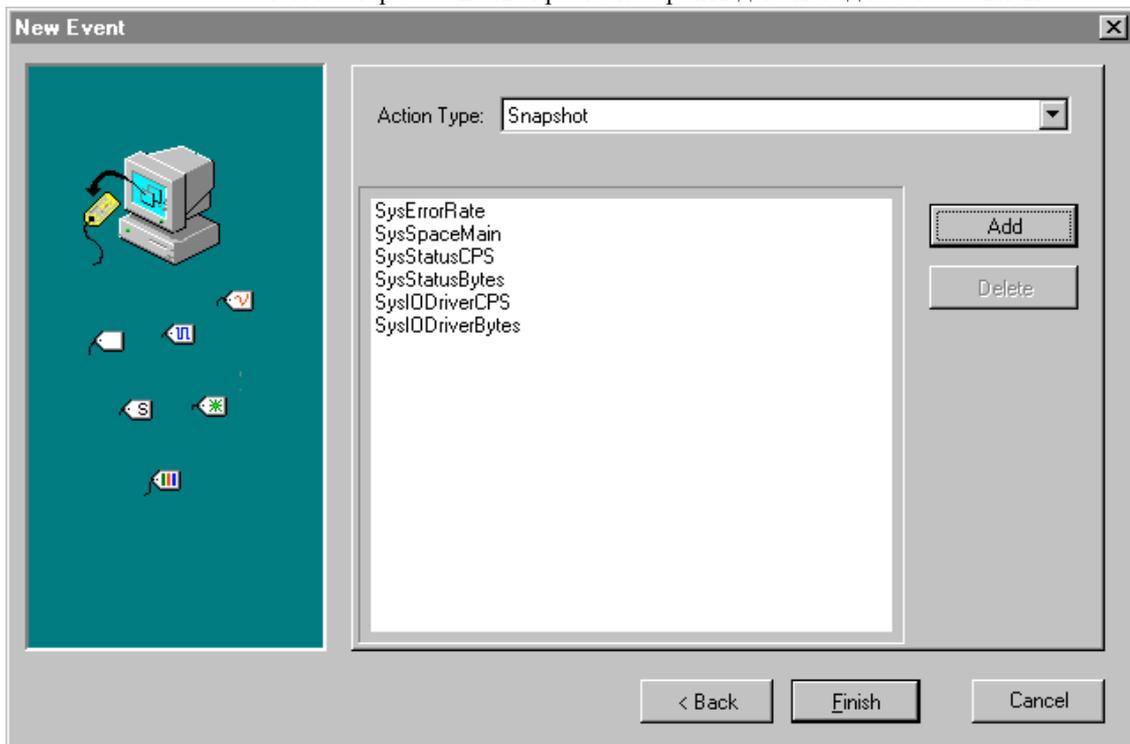
**Примечание.** Соблюдайте осторожность при присвоении временного интервала тэгам событий. См. "[Выделение системного потока для событий](#)" ранее в этой главе.

---

Подробнее о настройке детекторов см. соответствующий раздел далее в этой главе.

6. Нажмите **Next**.

Затем мастер **New Event** просит настроить действие для тэга события.



7. Настройте действие для тэга события. Действия могут быть none (нет), generic SQL (общее SQL), snapshot (снимок), e-mail (электронное сообщение), deadband (метвая зона) и summary (сводка). Нижняя часть панели свойств изменится в зависимости от выбранного типа действия.

Подробнее о настройке действий см. соответствующий раздел далее в этой главе.

8. Нажмите **Finish**.

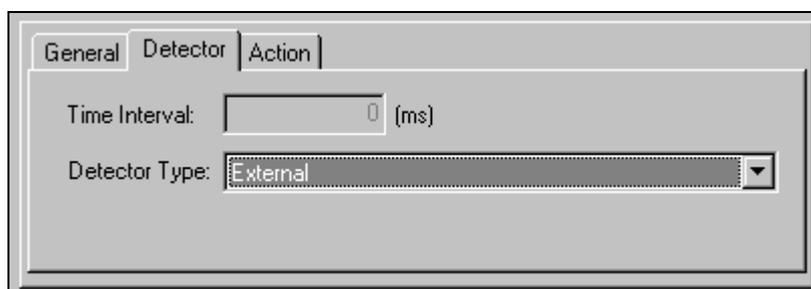
## Выбор внешнего детектора

Внешний детектор включается при помощи элемента управления ActiveX IndustrialSQL - ActiveEvent. Например, сценарий InTouch может включить событие IndustrialSQL используя методы ActiveEvent, которые очень близки к функциям. Внешний детектор нельзя настроить из InSQL Configure.

Подробнее об использовании элемента ActiveEvent см. в *Руководстве по элементам ActiveX IndustrialSQL Server*.

### ➤ Как выбрать внешний детектор:

1. Выберите в браузере тэг события.
2. Выберите вкладку свойств **Detector**.



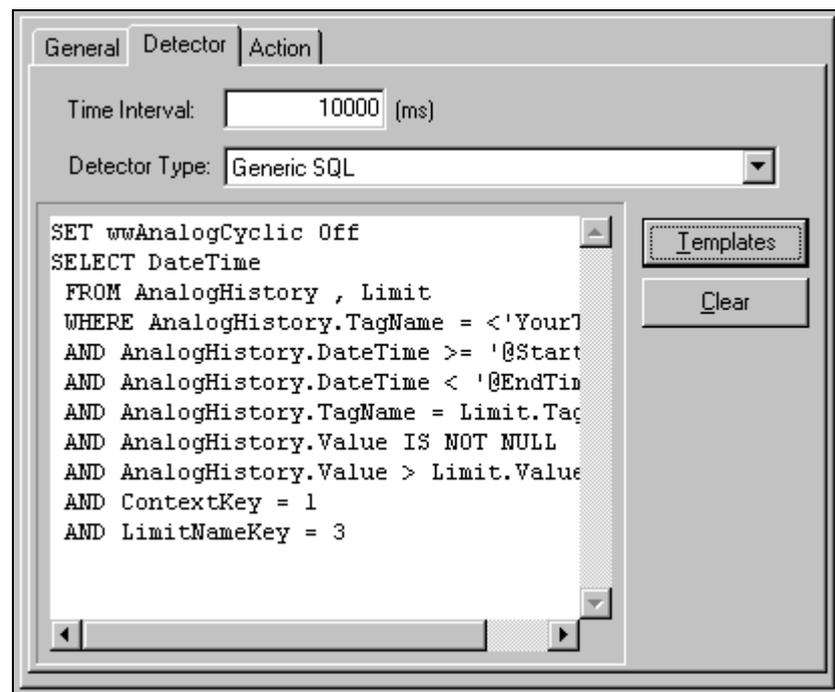
3. Поле **Time Interval** показывает интервал в миллисекундах, с которым оцениваются условия события. Если условия события удовлетворены, тогда событие сохраняется и все связанные с ним действия включаются. При использовании внешнего детектора это значение по умолчанию устанавливается на 0.
4. В списке **Detector Type** выберите **External**.
5. Для сохранения изменений нажмите кнопку . Или же в меню **File** выберите **Save**.
6. Для отмены изменений нажмите кнопку . Или же в меню **File** выберите **Undo**.

## Настройка общего детектора SQL

Общий детектор SQL фиксирует события на основе критериев, указанных в выражении SQL. Можно использовать заранее настроенные шаблоны SQL, которые хранятся в базе данных, как основы для сценариев или можно создать свой собственный сценарий абсолютно "с нуля".

### ➤ Как настроить общий детектор SQL:

1. Выберите в браузере тэг события.
2. Выберите вкладку свойств **Detector**.



3. В поле **Time Interval** введите интервал в миллисекундах, с которым оцениваются условия события. Если условия события удовлетворены, тогда событие сохраняется и все связанные с ним действия включаются. Временной интервал должен быть более 500 миллисекунд.

---

**Примечание.** Соблюдайте осторожность при присвоении временного интервала тэгам событий. См. "[Выделение системного потока для событий](#)" ранее в этой главе.

---

4. В списке **Detector Type** выберите **Generic SQL**.
5. В окне введите запрос, который должен фиксировать событие.

Для доступа к списку шаблонов SQL выберите **Templates**. Появится диалоговое окно, в котором можно выбрать для запроса заранее настроенные шаблоны SQL.

[Подробнее об использовании переменных системы управления событиями в запросе см. в разделе "Переменные системы управления событиями" ранее в этой главе.](#)

Для очистки окна нажмите **Clear**.

6. Для сохранения результатов нажмите кнопку . Или же в меню **File** выберите **Save**.
7. Для отмены изменений нажмите кнопку . Или же в меню **File** выберите **Undo**.

## Пример общего детектора SQL - множественные условия

Далее показан пример сценария общего детектора SQL со множественными условиями. Детектор событий будет работать только первые 10 минут каждого часа и только, когда 'SysTimeSec' равен 26, 27 или 28. В этом примере также используются переменные системы управления событиями: *@StartTime* и *@EndTime*.

```
SET wwEdgeDetection None
```

```
SET wwAnalogCyclic On
```

```
SET wwResolution 1000
```

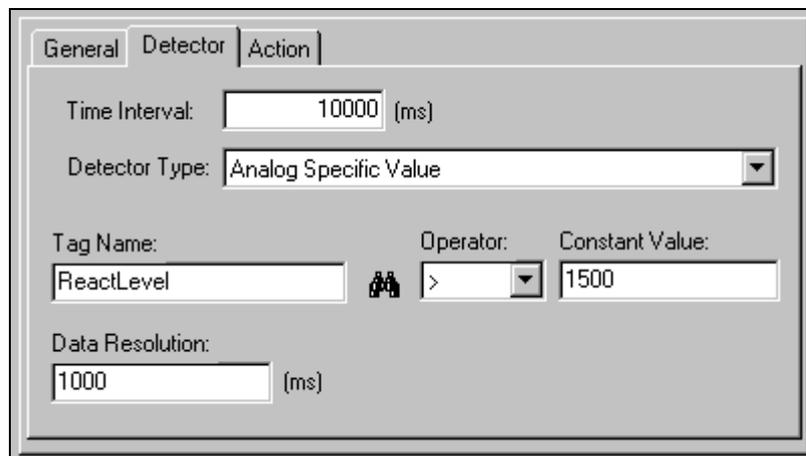
```
SELECT DateTime FROM AnalogWideHistory
WHERE SysTimeSec >= 26
      AND SysTimeSec <= 28
      AND SysTimeSec IS NOT NULL
      AND SysTimeMin < 10
      AND DateTime > "@StartTime"
      AND DateTime <= "@EndTime"
```

## Настройка детектора указанного аналогового значения

Детектор указанного аналогового значения фиксирует событие на основе критериев, связанных со значением конкретного аналогового тэга. Значение аналогового тэга связывается (с помощью операторов) с постоянным значением. Если значение совпадает с критерием, то событие регистрируется в таблице *EventHistory* и включаются все соответствующие действия.

➤ **Как настроить детектор указанного аналогового значения:**

1. Выберите в браузере тэг события.
2. Выберите вкладку свойств **Detector**.



3. В поле **Time Interval** введите интервал в миллисекундах, с которым оцениваются условия события. Если условия события удовлетворены, то событие сохраняется и включаются все соответствующие действия. Временной интервал должен быть более 500 миллисекунд.

---

**Примечание.** Соблюдайте осторожность при присвоении временного интервала тэгам событий. См. "[Выделение системного потока для событий](#)" ранее в этой главе.

---

4. В списке **Detector Type** выберите **Analog Specific Value**.
5. В поле **Tag Name** введите имя тэга, к которому относится критерий.

Для поиска по базе данных нажмите кнопку . Появится диалоговое окно **Tag Finder**, в котором можно подать в базу данных запрос на тэги. Подробнее см. раздел "[Поиск тэгов](#)" далее в этой главе.

6. В поле **Operator** выберите оператор для критерия.
7. В поле **Constant Value** введите значение, с которым будет сравниваться значение тэга при определении события. Введенное значение должно быть между минимальным и максимальным значением единицы измерения для этого значения.

8. В поле **Data Resolution** введите разрешение для данных.

Разрешение есть скорость (в мс) выборки данных из любой таблицы хранения IndustrialSQL Server. Например, при разрешении в 5000 мс будут выдаваться данные, сохраненные с интервалом в 5000 мс, начиная от стартовой даты и заканчивая конечной датой.

Разрешение данных применяется только, когда выбран параметр **NONE** для **Detect on the edge** в панели свойств **General**.

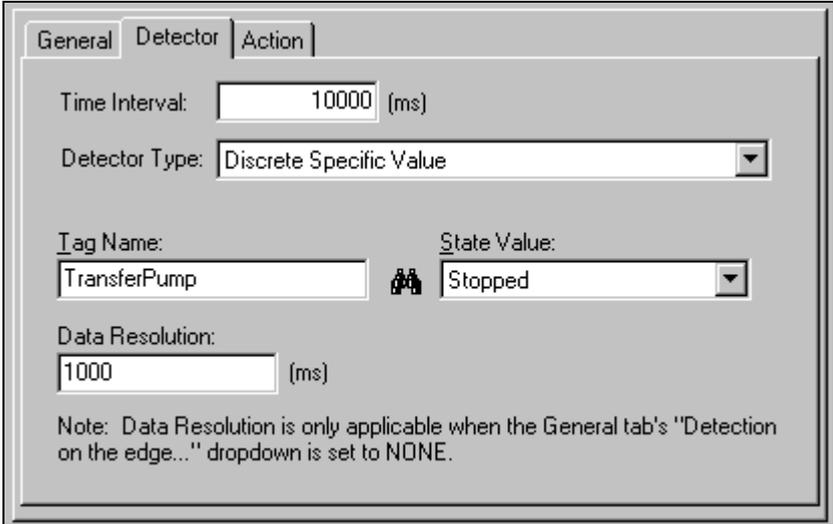
9. Для сохранения результатов нажмите кнопку . Или же в меню **File** выберите **Save**.
10. Для отмены изменений нажмите кнопку . Или же в меню **File** выберите **Undo**.

## Настройка детектора указанного дискретного значения

Детектор указанного дискретного значения фиксирует событие, если требуемое состояние равно текущему. Состояние дискретного тэга сравнивается с требуемым. Если состояния совпадают, событие регистрируется в таблице *EventHistory* и включаются все соответствующие действия.

### ➤ Как настроить детектор указанного дискретного значения:

1. Выберите в браузере тэг события.
2. Выберите вкладку свойств **Detector**.



General | **Detector** | Action

Time Interval:  (ms)

Detector Type:

Tag Name:   State Value:

Data Resolution:  (ms)

Note: Data Resolution is only applicable when the General tab's "Detection on the edge..." dropdown is set to NONE.

3. В поле **Time Interval** введите интервал в миллисекундах, с которым оцениваются условия события. Если условия события удовлетворены, тогда событие сохраняется и все связанные с ним действия включаются. Временной интервал должен быть более 500 миллисекунд.

---

**Примечание.** Соблюдайте осторожность при присвоении временного интервала тэгам событий. См. "[Выделение системного потока для событий](#)" ранее в этой главе.

---

4. В списке **Detector Type** выберите **Discrete Specific Value**.
5. В поле **Tag Name** введите имя тэга, для которого настраивается событие.

Для поиска по базе данных нажмите кнопку . Появится диалоговое окно **Tag Finder**, в котором можно подать в базу данных запрос на тэги. Подробнее см. раздел "[Поиск тэгов](#)" далее в этой главе.

6. В списке **State Value** выберите требуемое состояние дискретного тэга, которое повлечет за собой фиксацию события.
7. В поле **Data Resolution** введите разрешение для данных.

Разрешение есть скорость (в мс) выборки данных из любой таблицы хранения IndustrialSQL Server. Например, при разрешении в 5000 мс будут выдаваться данные, сохраненные с интервалом в 5000 мс, начиная от стартовой даты и заканчивая конечной датой.

Разрешение данных применяется только, когда выбран параметр **NONE** для **Detect on the edge** в панели свойств **General**.

8. Для сохранения результатов нажмите кнопку . Или же в меню **File** выберите **Save**.
9. Для отмены изменений нажмите кнопку . Или же в меню **File** выберите **Undo**.

## Настройка детектора планировщика

Детектор планировщика фиксирует событие в запланированное время по системным часам. Например, можно регистрировать событие в таблице *EventHistory* каждую неделю в понедельник в 14:00 и включать соответствующее событие, такое как подсчет минимального и максимального значения температуры, давления и уровня в Реакторе #7 и сохранения этого значения в базе данных. Также можно настроить систему так, чтобы каждую пятницу в 17:00 оператору производственного участка отправлялось электронное письмо с напоминанием о выключении оборудования.

### ➤ Как настроить детектор планировщика:

1. Выберите в браузере тэг события.
2. Выберите вкладку свойств **Detector**.

3. В списке **Detector Type** выберите **Schedule**.
4. В группе параметров **Frequency** укажите, как часто должно происходить событие.

После выбора частоты становятся доступными группы параметров **Set event detection time** и **Set event detection period**.

5. В группах **Set event detection time** and **Set event detection period** укажите время, соответствующее выбранной частоте.

Например, если требуется, чтобы событие происходило в 8:59 в первый день каждого месяца, следует выбрать частоту **Monthly** и настроить параметры **Set event detection time** следующим образом:

Или же, если требуется, чтобы событие происходило каждые три дня, начиная с 1 мая 1999 года, следует выбрать частоту **Periodic** и настроить параметры **Set event detection period** следующим образом:

- Для сохранения результатов нажмите кнопку . Или же в меню **File** выберите **Save**.
- Для отмены изменений нажмите кнопку . Или же в меню **File** выберите **Undo**.

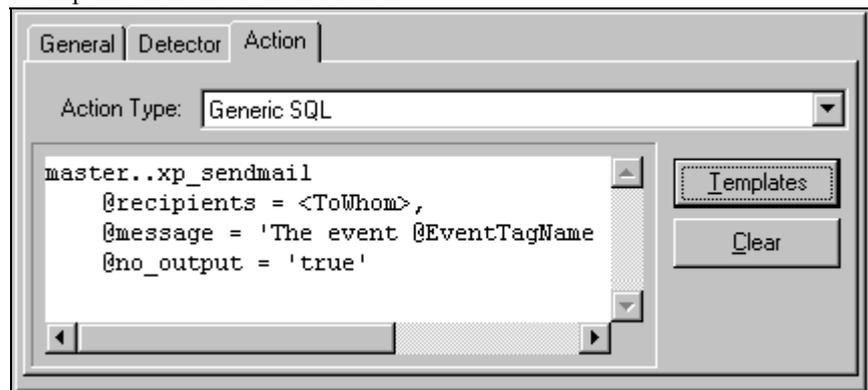
## Настройка общего действия SQL

Общее действие SQL включает действие, указанное в выражении SQL. Можно использовать заранее настроенные шаблоны SQL, хранящиеся в базе данных, как основу для сценария или можно создать собственный сценарий полностью "с нуля".

Нельзя подавать в IndustrialSQL Server более одного запроса на одно действие. Также для системы управления событиями не допустимы выражения GO. Рекомендуется проверять запросы SQL в SQL Server Query Analyzer перед их использованием в общих действиях SQL.

### ➤ Как настроить общее действие SQL:

- Выберите в браузере тэг события.
- Откройте панель свойств **Action**.



- В списке **Action Type** выберите **Generic SQL**.
- В окне введите запрос.

Для доступа к списку шаблонов SQL выберите **Templates**. Появится диалоговое окно, в котором можно выбрать заранее настроенные шаблоны SQL для запроса.

Подробнее об использовании переменных системы управления событиями в запросе см. в разделе "Переменные системы управления событиями" ранее в этой главе.

Для очистки окна нажмите **Clear**.

- Для сохранения результатов нажмите кнопку . Или же в меню **File** выберите **Save**.
- Для отмены изменений нажмите кнопку . Или же в меню **File** выберите **Undo**.

Следующие разделы описывают использование некоторых общих шаблонов SQL.

## Шаблоны общего действия SQL для электронного сообщения

Для настройки общего выражения SQL, посылающего электронное сообщение, выберите один из параметров " Send an E-mail message..." в списке шаблонов общих действий SQL. Например:

```
master..xp_sendmail
  @recipients = <ToWhom>,
  @message = 'Событие @EventTagName произошло в @EventTime',
  @no_output = 'true'
```

Во второй строке синтаксиса замените <ToWhom> на подпись e-mail или полный электронный адрес нужного получателя или получателей. Введите текст сообщения в переменной @message. Получатель(и) и само сообщение должны быть заключены в одинарные кавычки. Например:

```
master..xp_sendmail
  @recipients = 'John Doe',
  @message = 'Нужно проверить',
  @query = 'SELECT TagName,DateTime FROM EventHistory
           WHERE TagName = "SysStatusEvent"'
```

Для нормальной работы хранимой процедуры **xp\_sendmail** следует правильно настроить компоненты Microsoft SQL Server и SQL Mail. [Подробнее см. в разделе "Настройка действия с E-mail" далее в этой главе.](#)

## Шаблоны общего действия SQL для выполнения команды

Для настройки общего выражения SQL, выполняющего команду, выберите параметр "Invoke an External Application" в списке шаблонов общих действий SQL:

```
master..xp_cmdshell '<ваша команда>', no_output
```

В синтаксисе замените <ваша команда> на нужную команду. Пропустите no\_output если вам не нужна выдача результатов. Удостоверьтесь, что команда заключена в одинарные кавычки. Например:

```
master..xp_cmdshell 'dir *.exe'
```

Расширенная хранимая процедура **xp\_cmdshell** не запустит приложение Windows. Можно выполнять только простейшие команды DOS, пакетные или исполнимые файлы (.EXE), которые не открываются в пользовательском окне.

Для правильной отработки расширенной хранимой процедуры **xp\_cmdshell** следует установить необходимые права допуска. См. подробности в электронной документации по Microsoft SQL Server.

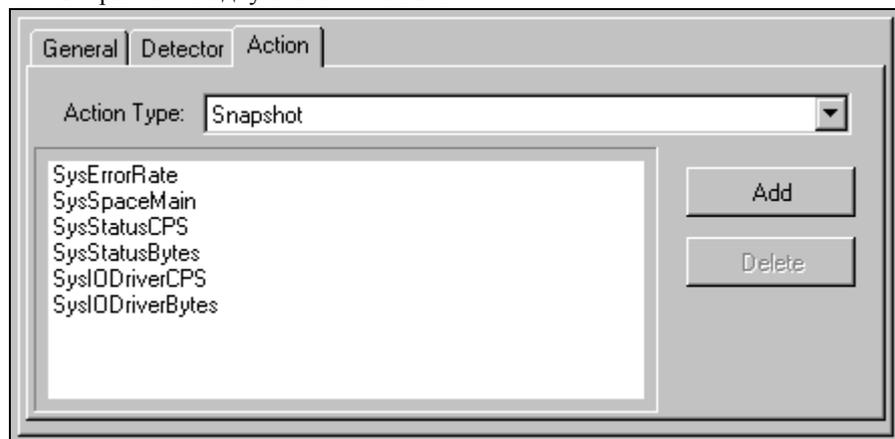
## Настройка снимков

Снимок записывает значения выбранного набора аналоговых, дискретных и текстовых тэгов во время возникновения события.

**Примечание.** Регистрацию событий невозможно отключить, если настроен снимок.

### ➤ Как настроить снимок:

1. Выберите в браузере тэг события.
2. Откройте вкладку свойств **Action**.



3. В списке **Action Type** выберите **Snapshot**.

В окне находится список всех тэгов, которые будут включены в снимок. В список могут быть включены аналоговые, дискретные или текстовые снимки.

4. Для добавления тэга нажмите **Add**. Появится диалоговое окно **Tag Finder**, в котором можно подать в базу данных запрос на тэги. Подробнее см. в разделе "**Поиск тэгов**" далее в этой главе.
5. Для удаления тэга выберите его в окне и нажмите **Delete**.
6. Для сохранения результатов нажмите кнопку . Или же в меню **File** выберите **Save**.
7. Для отмены изменений нажмите кнопку . Или же в меню **File** выберите **Undo**.

## Настройка электронного сообщения (E-Mail)

Действие типа электронного сообщения посылает заранее настроенное электронное сообщение при возникновении события. Для поддержки электронной почты системой управления событиями следует правильно настроить учетную запись на SQL Server, параметры почтового клиента Microsoft Outlook, функции SQL для работы с электронной почтой Microsoft SQL Server, и параметры доставки электронной почты из системы управления событиями.

### Настройка Microsoft SQL Server

Для нормальной работы Microsoft SQL с электронной почтой необходимо настроить Microsoft SQL Server на учетную запись пользователя, у которой есть действующий почтовый профиль MAPI. Эти шаги должны быть выполнены на компьютере, на котором запущена служба Microsoft SQL Server, обрабатывающая события электронной почты.

1. В Панели управления дважды нажмите на ярлык **Services**.

Появится диалоговое окно **Services**.

2. Выберите в списке **MSSQLServer** и нажмите **Startup**.

Появится диалоговое окно **Service**.

**Ошибка! Указано неверное имя файла.**

3. В группе параметров **Startup Type** выберите **Automatic**.

4. В группе **Log On As** выберите **This Account**. Введите учетную запись пользователя (общую или локальную) или нажмите кнопку с многоточием для поиска нужной учетной записи. Выбранная учетная запись должна иметь действующий почтовый профиль MAPI.

Подробнее об определении почтового профиля MAPI см. в разделе "Определение почтового профиля Microsoft Outlook" далее в этой главе.

5. В полях **Password** и **Confirm Password** введите пароль для учетной записи пользователя.
6. Нажмите **OK**.
7. Если это еще не сделано, выберите службу MSSQL Server в диалоговом окне **Services** и нажмите **Stop** для остановки службы.
8. После остановки службы MSSQL нажмите **Start** для перезапуска службы.
9. Нажмите кнопку **Close**.

### Определение почтового профиля Microsoft Outlook

Для правильной работы с электронной почтой система Microsoft SQL Server должна знать почтовый профиль MAPI для учетной записи MSSQL Server. Следует определить имя профиля MAPI на том компьютере, где запущен MSSQL Server:

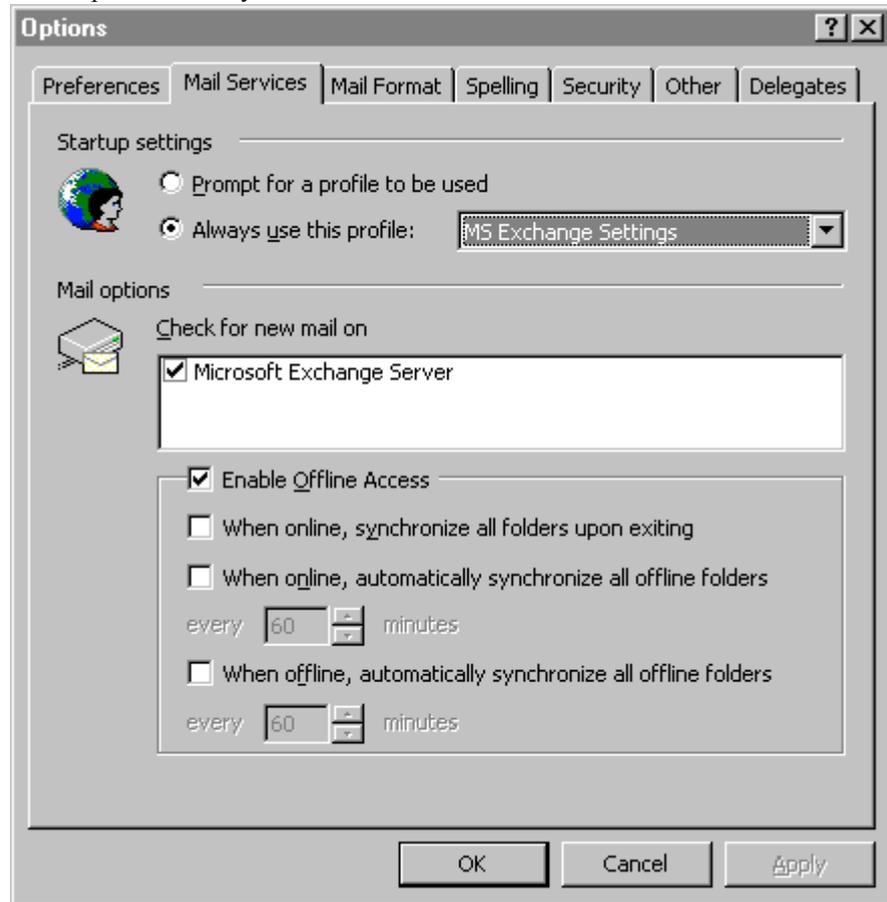
➤ **Как определить профиль MAPI:**

1. Запустите Microsoft Outlook.

2. В меню **Tools** выберите **Options**.

Появится диалоговое окно **Options**.

3. Откройте вкладку **Mail Services**.



4. Определите профиль MAPI, открыв ниспадающий список справа от параметра **Always use this profile**.

Обычно почтовый профиль установлен на "MS Exchange Settings" для использования пользовательского профиля на сервере Exchange.

5. Нажмите **OK**.

Подробнее о настройке пользовательских учетных записей для клиента Exchange см. в документации Microsoft.

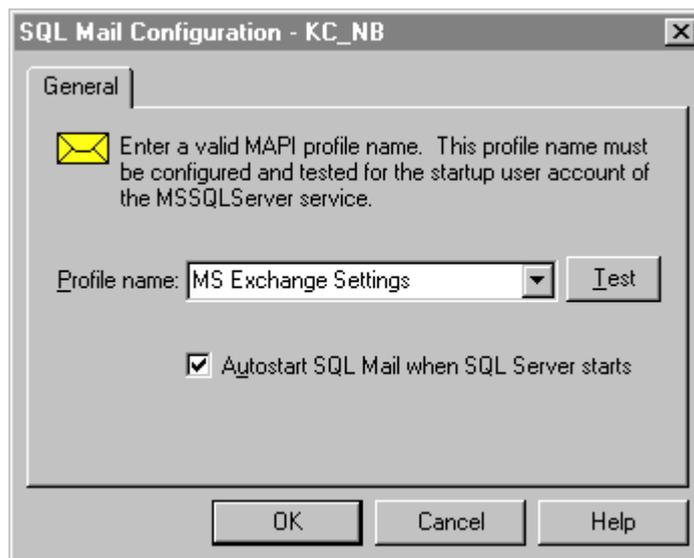
## Настройка почты SQL в SQL Enterprise Manger

Почта SQL настроена на использование SQL Server Enterprise Manager. Должен быть запущен Microsoft SQL Server.

### ➤ Как настроить почту SQL:

1. Запустите SQL Server Enterprise Manager.
2. Разверните нужный сервер и откройте папку **Support Services**.
3. Нажмите правой кнопкой на **SQL Mail**. В контекстном меню выберите **Properties**.

Появится диалоговое окно **SQL Mail Configuration**.



4. В поле **Profile name** введите имя профиля MAPI.  
Об определении имени профиля см. в разделе "Определение имени почтового профиля Microsoft Outlook" ранее в этой главе.
5. Для автоматического запуска почты SQL нажмите **Autostart SQL Mail when SQL Server starts**.

Если этот параметр не выбран, почта SQL должна перезапускаться вручную после каждого перезапуска Microsoft SQL Server.

6. Для проверки имени профиля нажмите **Test**.

Если проверка успешна, то будет получено следующее сообщение: "Successfully started (and stopped) a MAPI session with this profile". Если получено сообщение об ошибке, удостоверьтесь, что вы используете правильное имя профиля, соответствующее учетной записи службы MSSQL Server.

Подробнее см. в разделе "Настройка Microsoft SQL Server" ранее в этой главе.

7. Нажмите **OK** для возврата в окно SQL Server Enterprise Manager.
8. Нажмите правую кнопку на папке **SQL Mail**. Когда появится меню ярлычков, нажмите **Start**.

Если почта SQL успешно запущена, появится информационное сообщение. Индикатор состояния почты SQL станет зеленым, что указывает на то, что почта SQL запущена:



## Настройка электронной почты (E-Mail)

---

**Примечание.** Можно использовать только адреса Microsoft Outlook. Адреса Интернет прямо не поддерживаются.

---

- **Как настроить электронное сообщение:**

1. Выберите в браузере тэг события.
2. Откройте панель свойств **Action**.

**Ошибка! Указано неверное имя файла.**

3. В списке **Action Type** выберите **E-mail**. Появится стандартный интерфейс электронной почты.
4. В строке **To** введите адрес электронной почты Outlook, куда будет послано электронное сообщение после возникновения события. При желании можно послать копию электронного письма одному или нескольким лицам, указав их адреса в строке **Cc**.

Нажав на кнопку **To** или **Cc**, можно открыть адресную книгу.

5. В строке **Subject** введите тему сообщения.

Строку **Subject** нельзя оставлять пустой.

6. Введите текст электронного сообщения в пустом окне.

7. Для сохранения результатов нажмите кнопку . Или же в меню **File** выберите **Save**.

8. Для отмены изменений нажмите кнопку . Или же в меню **File** выберите **Undo**.

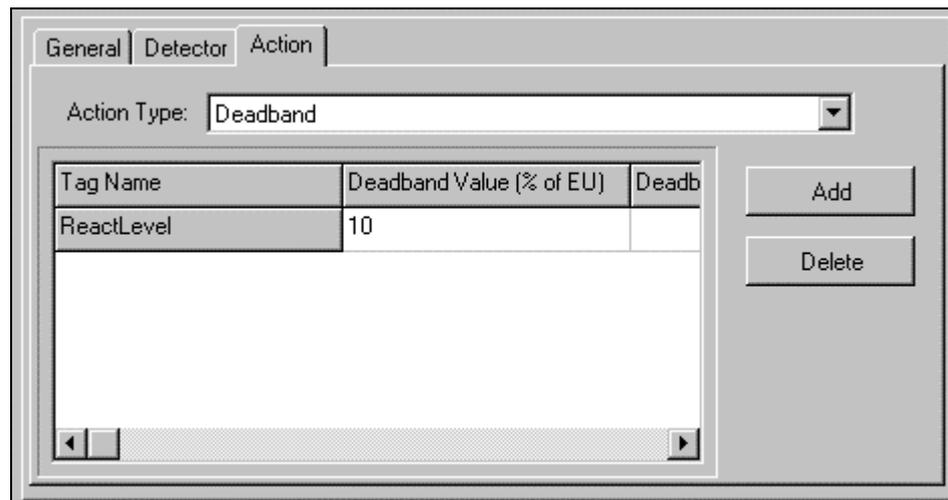
## Настройка изменения мертвой зоны

Действие типа изменение мертвой зоны изменяет время и/или значение мертвой зоны хранения для одного или нескольких тэгов. Мертвые зоны используются алгоритмом дельта-хранения (сохранение методом исключения) для определения того, когда следует сохранять значение конкретного тэга.

Изменения мертвой зоны используются для увеличения объема хранимых данных при возникновении определенного события. Например, если котел выключился, можно начать сохранять значения конкретного тэга с более высокой скоростью для определения причины выключения.

➤ **Как настроить изменение мертвой зоны:**

1. Выберите в браузере тэг события.
2. Откройте панель свойств **Action**.



3. В списке **Action Type** выберите **Deadband**.
4. Для выбора тэга, для которого устанавливается новая мертвая зона, нажмите **Add**. Появится диалоговое окно **Tag Finder**, в котором можно подать в базу данных запрос на тэги. Подробнее см. в разделе "**Поиск тэгов**" далее в этой главе.
5. В поле **Deadband Value** введите новое процентное значение в единицах измерения.

Значение мертвой зоны есть процент от полного масштаба в единицах измерения, на который значение тэга должно измениться до того, как оно будет обновлено в базе данных. Значение мертвой зоны применяется только при дельта-хранении (сохранение методом исключения). Если этот атрибут получает значение 0 или NULL, мертвая зона не используется.

6. В поле **Deadband Time** введите новое время в миллисекундах.

Временная мертвая зона есть минимальное время в миллисекундах между сохранением значений для одиночного тэга. Тэг, значение которого изменяется более одного раза за этот период времени, будет игнорирован алгоритмом дельта-хранения. Мертвая зона равная 0 показывает, что система будет сохранять значение тэга при каждом его изменении.

7. Для удаления тэга из списка выберите тэг и нажмите **Delete**.

8. Для сохранения результатов нажмите кнопку . Или же в меню **File** выберите **Save**.

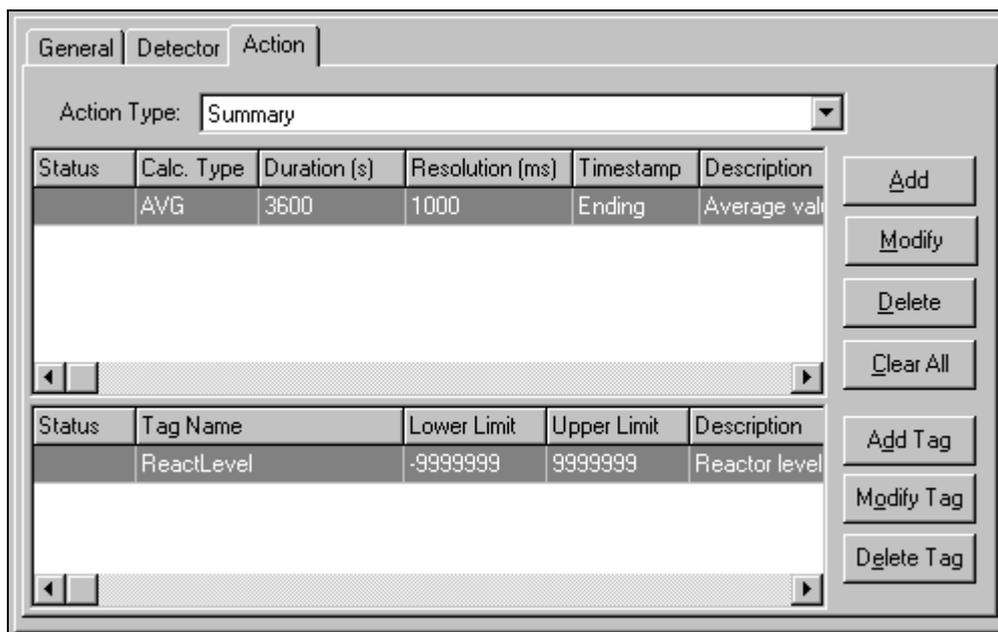
9. Для отмены изменений нажмите кнопку . Или же в меню **File** выберите **Undo**.

## Настройка создания сводок

Создание сводки есть набор агрегаций, проводимых со значениями выбранных тэгов в диапазоне между стартовым временем и конечным временем с указанным разрешением. При настройке действия сводки следует указать тип агрегации, которую нужно произвести (операция сводки), и аналоговый тэг, для которого создается сводка. Когда создание сводки выполняется при запуске, результаты сохраняются в таблицах *SummaryHistory* и *SummaryData* в базе данных *Runtime*.

### ➤ Как настроить создание сводки:

1. Выберите в браузере тэг события.
2. Откройте вкладку свойств **Action**.



3. В списке **Action Type** выберите **Summary**.
4. Для добавления новой операции сводки нажмите **Add**.  
См. "Добавление операции сводки".  
Новая операция сводки появится в верхней таблице панели свойств.
5. Для присвоения тэга операции сводки выберите операцию сводки и нажмите **Add Tag**.  
См. "Добавления тэга сводки".  
Для одной операции сводки можно добавить несколько тэгов.
6. Для очищения таблицы от всех операций сводок нажмите **Clear All**.
7. Для сохранения результатов нажмите кнопку . Или же в меню **File** выберите **Save**.
8. Для отмены изменений нажмите кнопку . Или же в меню **File** выберите **Undo**.

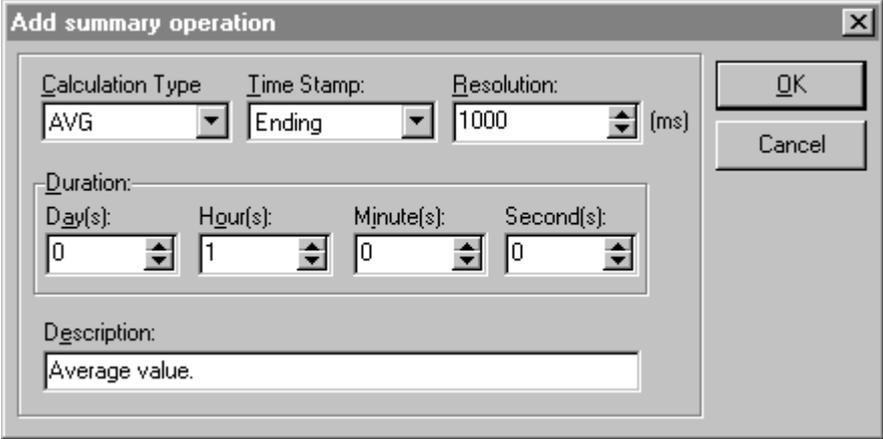
Значения тэгов с плохим качеством не обрабатываются до проведения агрегации. Для проведения агрегации только с хорошим качеством, например, выполните запрос агрегации с таблицей *AnalogHistory*, где значение столбца *Quality* равняется 0.

## Добавление операции сводки

Можно добавить несколько операций сводок для одного действия сводки, если разные операции сводок не имеют одинаковых настроек.

1. В панели свойств создания сводки нажмите **Add**.

Появится диалоговое окно **Add Summary Operation**.



2. В списке **Calculation Type** выберите тип агрегации: среднее (AVG), сумма (SUM), минимальное значение (MIN), максимальное значение (MAX).
3. В списке **Time Stamp** выберите для подсчитываемых данных метку начального времени или метку конечного времени периода подсчета.

Подсчитываемые данные сохраняются с учетом выбранной метки времени.

4. В поле **Resolution** введите разрешение выборки для которых создается сводка.

Обычно, чем больше разрешение, тем точнее результат, потому что в агрегацию включается больше значений; зато подсчет займет больше времени и потребует больше ресурсов сервера. Избегайте слишком большого разрешения для создания сводок, связанных с детекторами-планировщиками, которые охватывают большой временной диапазон, такие как еженедельные.

Разрешение также полезно при подсчете суммы. Например, установка разрешения в 60000 миллисекунд для потока в галлонах/минутах автоматически выдаст результат полного объема.

5. В поле **Duration** введите период, за который должен быть проведен подсчет.

Например, если создание сводки настроено на длительность в 1 час с детектором, установленным на 3:00 каждого понедельника, система будет проводить агрегацию со значениями, сохраненными по понедельникам с 2:00 до 3:00.

6. В поле **Description** введите описание операции сводки.
7. Закончив определение операции сводки, нажмите **OK**.

Новая операция сводки теперь появится в верхней таблице панели свойств создания сводки.

## Удаление операции сводки

➤ **Как удалить операцию сводки:**

1. Выберите операцию в таблице.
2. Нажмите **Delete**.

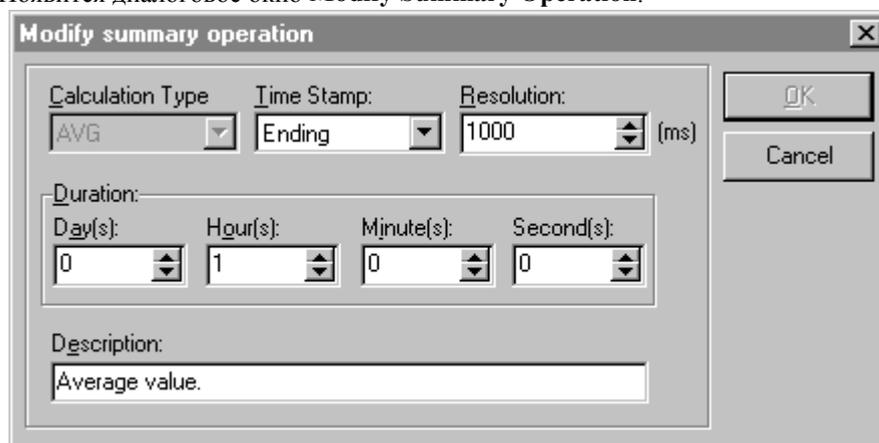
## Изменение операции сводки

При изменении операции сводки можно заметить несоответствие между старыми данными сводки и новыми.

➤ **Как изменить существующую операцию сводки:**

1. Выберите операцию сводки в таблице.
2. Нажмите **Modify**.

Появится диалоговое окно **Modify Summary Operation**.



3. Это диалоговое окно идентично диалоговому окну для добавления операций сводок за исключением того, что поле **Calculation Type** в нем недоступно.

Измененную операцию сводки нельзя сохранить, если ее критерий идентичен существующей операции сводки, связанной с текущим тэгом события.

Подробнее об этом диалоговом окне см. в разделе "Добавление операции сводки" ранее в этой главе.

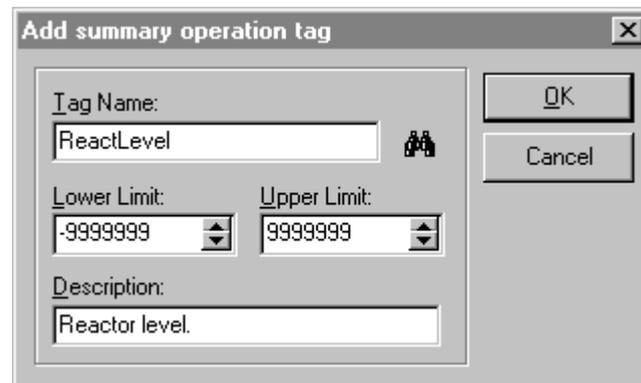
4. После того, как изменения сделаны, нажмите **OK**.

## Добавление тэга в сводку

В одну операцию сводки можно добавить несколько тэгов.

1. В панели свойств создания сводки выберите **Add Tag**.

Появится диалоговое окно **Add Summary Operation Tag**.



2. В поле **Tag Name** введите имя тэга.

Для поиска тэга по базе данных нажмите кнопку . Появится диалоговое окно **Tag Finder**, в котором можно подать в базу данных запрос на тэги.

[Подробнее см. в разделе "Поиск тэгов" далее в этой главе.](#)

3. Установите границы допустимых значений в полях **Lower Limit** и **Upper Limit**.

Установка границ допустимых значений позволяет контролировать диапазон значений, между которыми будет проводиться подсчет. Значения, выходящие за пределы этих границ игнорируются в подсчетах. По умолчанию эти значения установлены на очень высокие и низкие, так что обрабатываются все значения.

Например, если нижняя граница установлена на 1000, алгоритм подсчета игнорирует все выдаваемые данные со значением менее 1000.

4. В поле **Description** введите описание для этого тэга сводки.
5. Нажмите **OK**.

Новый тэг сводки появится в нижней таблице панели свойств создания сводки.

## Удаление тэга сводки

### ➤ Как удалить тэг сводки:

1. Выберите тэг в таблице.
2. Нажмите **Delete**.

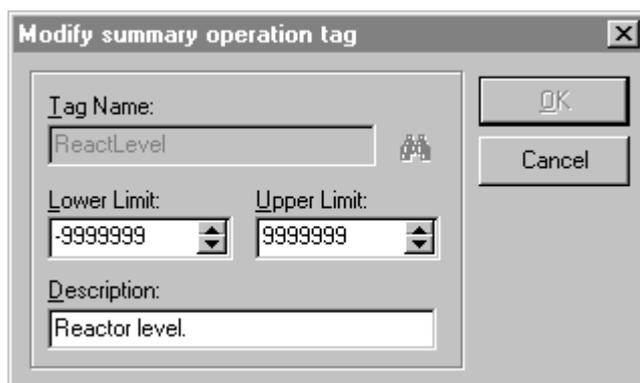
## Изменение тэга сводки

При изменении тэга сводки можно заметить несоответствие между старыми данными сводки и новыми.

➤ **Как изменить существующий тэг сводки:**

1. Выберите тэг в таблице.
2. Нажмите **Modify**.

Появится диалоговое окно **Modify Summary Operation Tag**.



3. Это диалоговое окно идентично диалоговому окну для добавления операций сводок за исключением того, что поле **Tag Name** недоступно. [Подробнее об этом диалоговом окне см. в разделе "Добавление тэга в сводку" ранее в этой главе.](#)
4. После внесения изменений нажмите **OK**.

## Поиск тэгов

Искать тэги в базе данных можно при помощи диалогового окна **Tag Finder**.

Это диалоговое окно можно вызвать, нажав кнопку  на панели свойств или нажатием кнопки **Add** при заполнении списка тэгов.

При помощи **Tag Finder** можно быстро найти в базе данных тэги, удовлетворяющие указанным параметрам поиска по имени или описанию тэга. Также можно искать тэги методом "выбрать и нажать" или путем ввода выражения SQL. Если **Tag Finder** выдает набор тэгов, удовлетворяющих запросу, можно выбрать те, которые требуется включить.

## Использование панели формирования запроса

Панель **Form Query** диалогового окна **Tag Finder** используется для выбора критериев поиска в базе данных.

The screenshot shows the 'Tag Finder' dialog box with the 'Form Query' tab selected. The 'Tag Name' field is set to 'Ends with' and 'level'. The 'Operator' is 'And'. The 'Description' is 'All'. Under 'Tag Types', 'Analog' is checked. The 'Found Tags' list contains '\$AccessLevel' and 'ProdLevel'. The 'Target Tags' list contains 'ReactLevel'. The status bar at the bottom shows 'Found Tags: 2' and 'Target Tags: 1'.

### ➤ Как сформировать и выполнить запрос:

1. В списке **Tag Name** выберите фразу для критерия поиска имени тэга. Например, "Ends with" (кончается на).
2. Укажите в запросе параметры поиска имени тэга. Например, "level" (уровень). При поиске тэгов нужны только символы подстановки для игнорирования средней части поискового слова. Например, "ур%ень".

Для игнорирования параметра поиска нажмите **Not**.

3. Для добавления параметров поиска по описанию тэга выберите логический оператор из списка **Operator**.
4. В списке **Description** выберите фразу для критерия поиска описания тэга. Это поле необязательно для заполнения.
5. Введите в запрос параметры поиска описания тэга. Это поле необязательно для заполнения. При поиске тэгов нужны только символы подстановки для игнорирования средней части описания.

Для игнорирования параметра поиска нажмите **Not**.

6. В группе **Tag Types** выберите группу тэгов для поиска.
7. После установки всех параметров запроса нажмите **Find Now** для выполнения запроса.

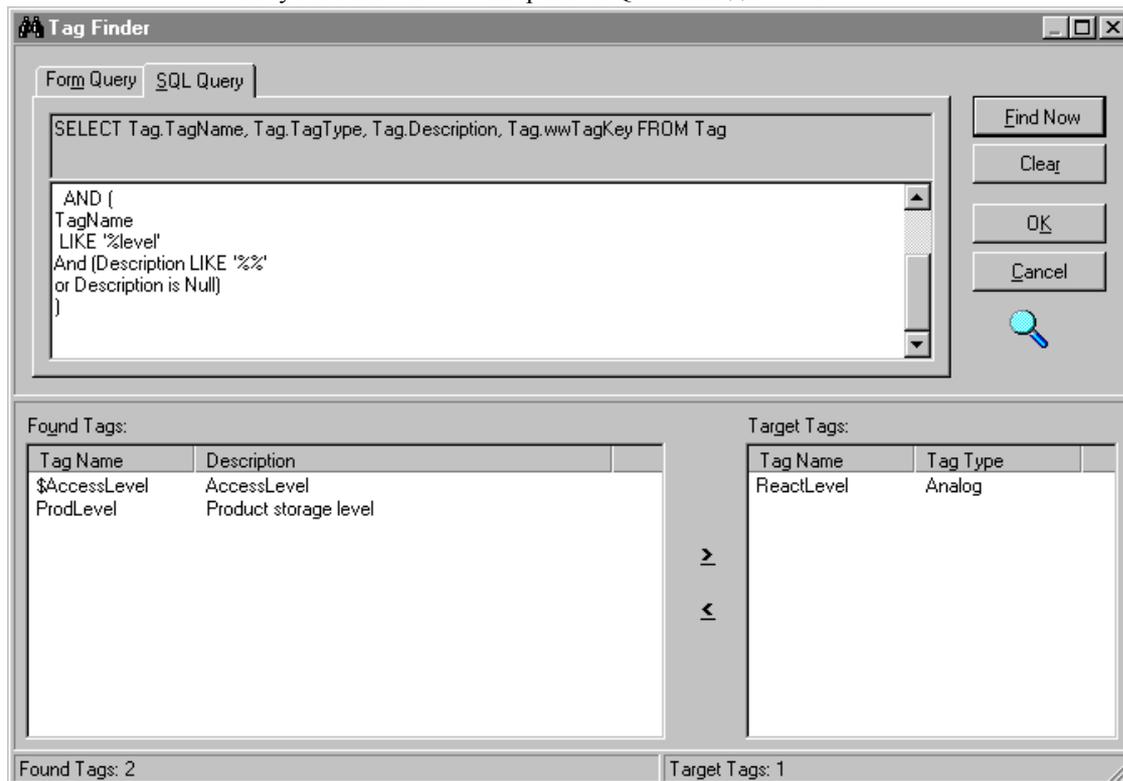
Результаты поиска будут показаны в окне **Found Tags** в диалоговом окне **Tag Finder**.

8. Для добавления тэга выберите тэг в окне **Found Tags** и используйте кнопку со стрелкой, чтобы переместить выбранный тэг в окно **Target Tags**.
9. Нажмите **OK**.

Для просмотра синтаксиса запроса к базе данных откройте вкладку **SQL Query**.

## Использование панели SQL Query

Панель **SQL Query** диалогового окна **Tag Finder** используется для ввода и запуска собственных запросов SQL к базе данных.



### ➤ Как сформировать и выполнить запрос:

1. В окне запроса введите параметры выражения WHERE для запроса SQL.

---

**Примечание.** Изменить выражение SELECT невозможно; нужные таблицы и столбцы уже заполнены за вас.

---

2. После того, как введены все параметры запроса, нажмите **Find Now**.  
Результаты поиска будут показаны в окне **Found Tags** в диалоговом окне **Tag Finder**.
3. Для добавления тэга выберите тэг в окне **Found Tags** и используйте кнопку со стрелкой, чтобы переместить тэг в окно **Target Tags**.
4. Нажмите **OK**.

## Удаление тэга события

➤ **Как удалить тэг события:**

1. В браузере InSQL Configure выберите сервер, с которого вы хотите удалить тэг события.
2. Выберите в папке Events тэг события, который вы хотите удалить.
3. В меню **File** выберите **Delete**. Или нажмите кнопку  на панели инструментов.

## Настройка безопасности для ActiveEvent

Для того, чтобы элемент управления ActiveEvent работал, необходимо правильно настроить атрибуты безопасности (разрешение на запуск). На компьютере IndustrialSQL Server атрибуты безопасности настраиваются при помощи программы DCOMCNFG.EXE.

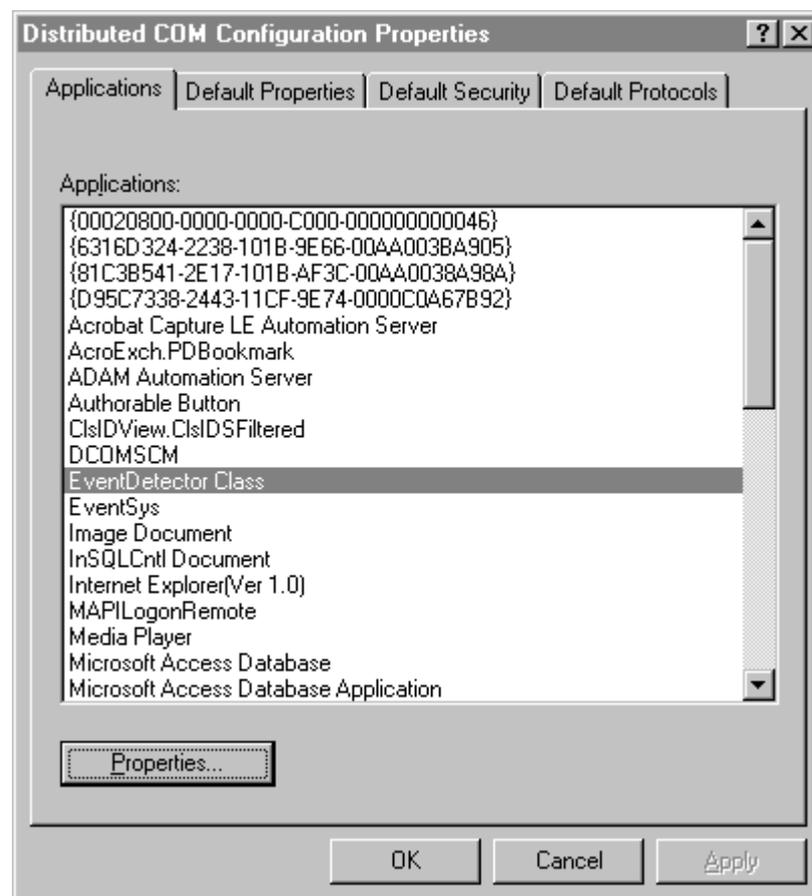
➤ **Как настроить атрибуты безопасности ActiveEvent:**

1. В меню **Start** выберите **Run**.

Появится диалоговое окно **Run**.

2. В поле **Open** наберите DCOMCNFG.EXE.
3. Нажмите **OK**.

Появится диалоговое окно **Distributed COM Configuration Properties**.

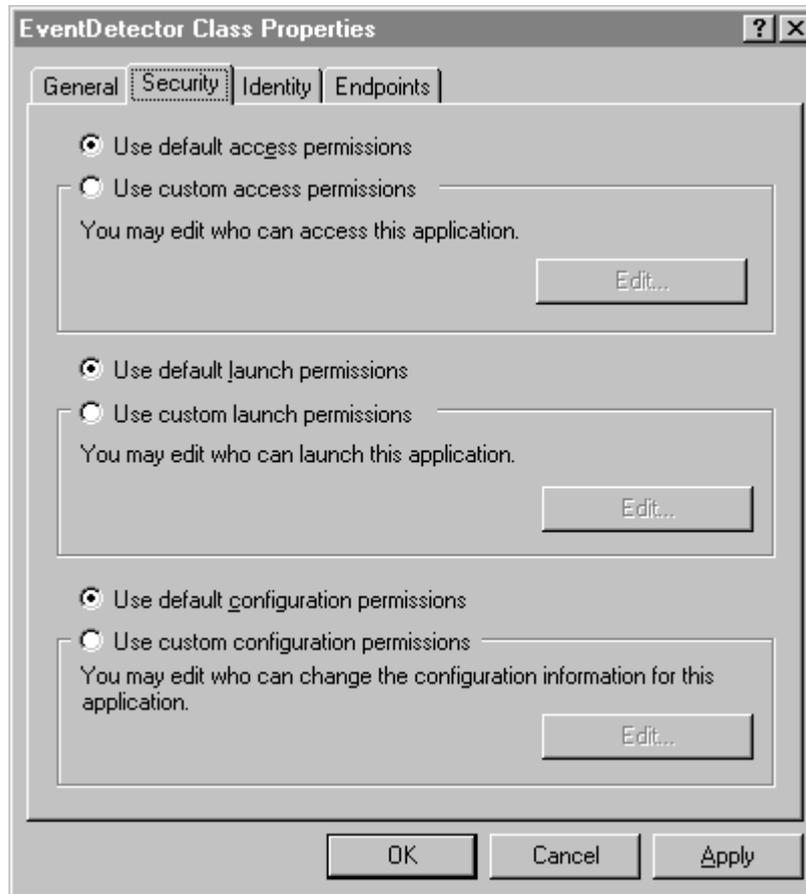


4. В списке **Applications** выберите **EventDetectorClass**.

5. Выберите **Properties**.

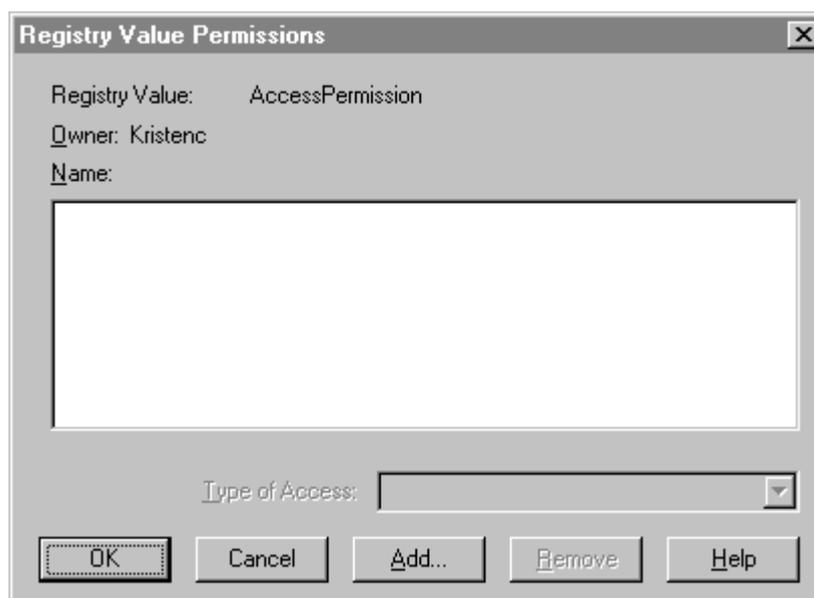
Появится диалоговое окно **EventDetector Class Properties**.

6. Откройте вкладку **Security**.



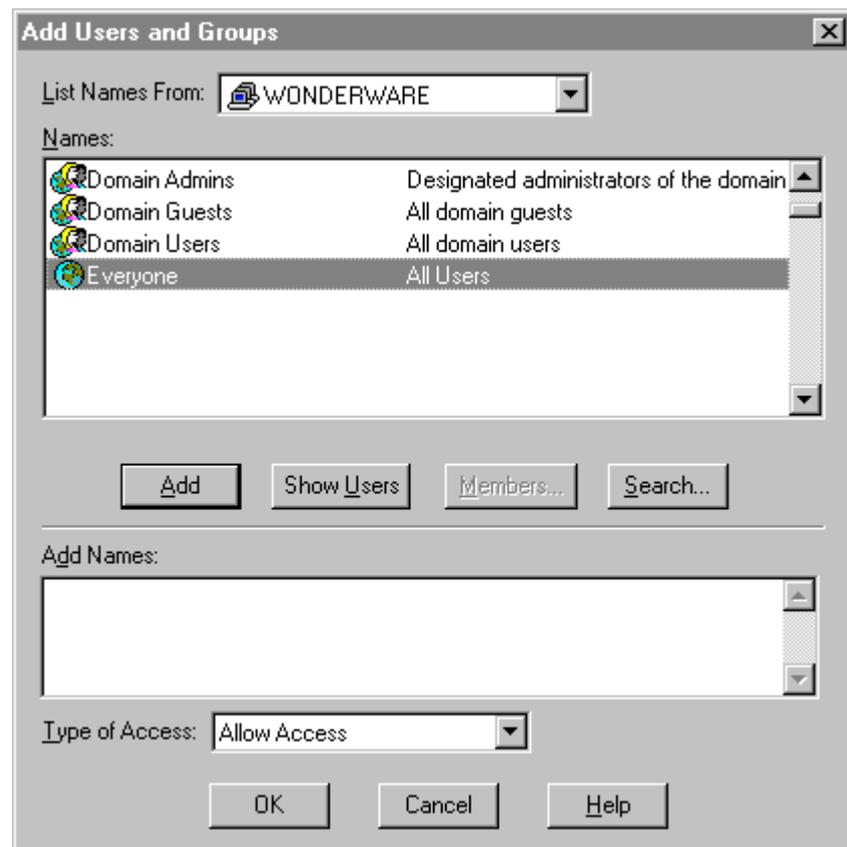
7. Выберите **Use custom access permissions** и затем нажмите **Edit**.

Появится диалоговое окно **Registry Value Permissions**.



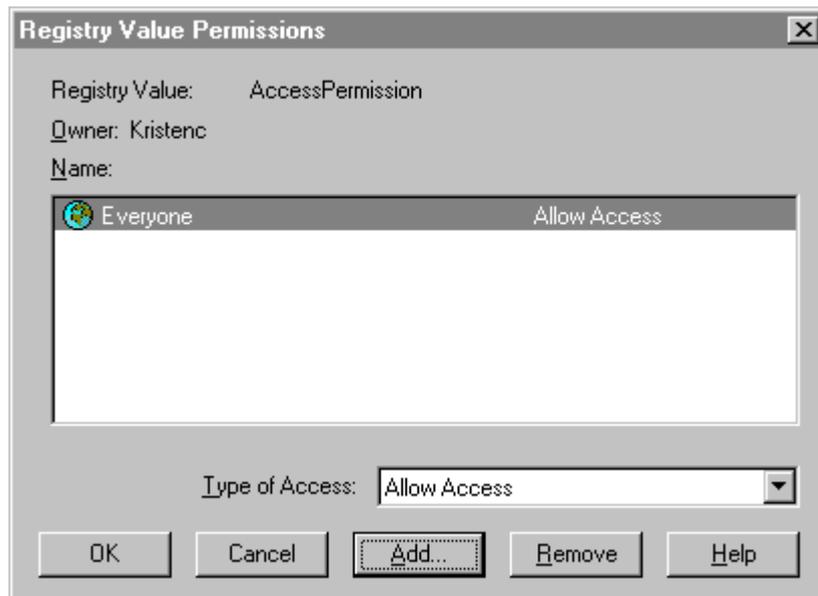
8. Нажмите **Add**.

Появится диалоговое окно **Add Users and Groups**.



9. В списке **Names** выберите группу **Everyone**.
10. Нажмите **Add**.
11. Выберите группу **Everyone** в списке **Add Names**.
12. В списке **Type of Access** выберите **Allow Access**. (Для Windows NT Workstation это **Full Control**).
13. Нажмите **OK**.

Теперь диалоговое окно **Registry Value Permissions** показывает группу **Everyone** с разрешенным доступом.



14. Нажмите **OK** для возврата в диалоговое окно **EventDetector Class Properties**.
15. Выберите **Use custom launch permissions** и затем нажмите **Edit**.
16. Повторяйте шаги с 8 по 14 для добавления группы **Everyone** в список пользователей, у которых есть разрешение на запуск. В списке **Type of Access** диалогового окна **Add Users and Groups** обязательно выберите **Allow Launch**.
17. Нажмите **OK** для закрытия диалогового окна **EventDetector Class Properties**.
18. Нажмите **OK** для закрытия программы DCOMCNFG.EXE.

## Выборка данных зарегистрированных событий

Когда событие зафиксировано, система управления событиями регистрирует следующее в таблице *EventHistory*: 1) имя тэга события, с которым связаны критерии; 2) метка даты/времени возникновения события; 3) время фиксации события и 4) информация о критериях фиксации.

Информация о критериях фиксации, показанная в столбце *Edge*, может быть интерпретирована следующим образом:

Значение	Описание
0	Ведомое определение края (детекторы SQL)
1	Ведущее определение края (детекторы SQL)
2	Определение на обоих краях (детекторы SQL)
3	Нет определения края (детекторы SQL)
4	Фиксация планировщиком
5	Внешний детектор

Если для события был настроен снимок, данные снимка регистрируются в таблице *SnapshotTag* и в таблице снимков для определенного типа тэгов (например, таблица *AnalogSnapshot*). Если для события настроено создание сводки, агрегированные данные сохраняются в таблицах *SummaryHistory* и *SummaryData*.

Для просмотра архива событий подается запрос в таблицу *EventHistory*. Например, тэг события, "EventTag1" был создан для определения момента, когда значение тэга "ReactLevel" равнялось 2000. Запрос на выборку архива событий выглядит следующим образом:

```
SELECT * FROM EventHistory
WHERE TagName = 'EventTag1'
      AND DateTime >= 'Apr 16 1999 12:36 PM'
      AND DateTime <= 'Apr 16 1999 12:41 PM'
```

Для просмотра информации снимка тэга события (символы подстановки не разрешены) используйте представление **v\_EventSnapshot** и укажите имя тэга события в выражении WHERE. Например:

```
SELECT * FROM v_EventSnapshot
WHERE Event = 'EventTag1'
```

По запросу будет выдано имя тэга события, время возникновения события и время его фиксации, а также имя, значение и качество каждого тэга в снимке. Для значения каждого тэга будет выдана одна строка.

# Мониторинг системы

Мониторинг системы позволяет оценить работоспособность и производительность системы IndustrialSQL Server в целом, а также проверить правильность работы системных драйверов и целостность системы.

В данной главе описывается, как вести наблюдение за системой с помощью программы Windows NT Performance Monitor, драйвера целостности системы и программы просмотра событий Windows NT Event Viewer.

## Содержание

- Оценка производительности системы
- Отслеживание драйверов системы хранения
- Отслеживание переменных целостности системы
- Производительность системы
- Журналы событий Windows NT
- Просмотр событий Windows NT

## Оценка производительности системы

Производительность системы IndustrialSQL Server можно рассматривать с двух точек зрения: эффективность процессов, выполняемые в среде операционной системы Windows NT, или эффективность приложений, выполняющих запись-чтение в базе данных. В данном разделе описывается оценка производительности системы IndustrialSQL Server с обеих точек зрения.

## Отслеживание драйверов системы хранения

Чтобы убедиться в том, что система IndustrialSQL Server работает с максимальной эффективностью, важно отслеживать производительность компонентов, отвечающих за хранение и обмен данных. Для запуска, остановки и наблюдения за системой IndustrialSQL Server используется программа InSQL Control. С ее помощью можно отслеживать данные, обрабатываемые всеми драйверами.

Используя программу, InSQL Control можно отслеживать статистику по каждому драйверу устройства, так и суммарную статистику по всем драйверам. Драйверы, называемые также источниками данных, включают в себя:

- Серверы ввода-вывода.
- Системный драйвер, отслеживающий переменные целостности системы IndustrialSQL Server.

Ниже приводятся типы показателей, описывающих состояние сервера:

Показатель	Описание
CPS	Значение счетчика символов в секунду.
Bytes	Количество полученных байт.
Seconds	Количество секунд с начала отсчета
Packets	Количество полученных пакетов
Nacks	Количество отвергнутых запросов.
Restarts	Количество перезапусков драйвера

Полное описание InSQL Control см. в главе 5 "InSQL Control" руководства администратора.

## Отслеживание переменных целостности системы

Общая целостность системы IndustrialSQL Server отслеживается системным драйвером SYSDRV.EXE. Этот драйвер отслеживает ключевые переменные системы IndustrialSQL Server и пересылает их текущие значения в систему хранения. Системный драйвер устанавливается автоматически вместе с IndustrialSQL Server и запускается вместе с системой хранения данных с помощью InSQL Control.

Эти системные переменные являются предопределенными в IndustrialSQL Server и становятся доступными сразу после запуска. Их значения обеспечивают достаточно полную картину состояния системы IndustrialSQL Server.

### Системные переменные

Ниже приводятся таблицы с описаниями системных переменных. Тэги могут быть аналоговые (A); дискретные (B); текстовые (S) или событийные (E). Текущее значение отслеживаемого тэга пересылается в систему хранения в соответствии с заданной частотой.

#### Общие переменные

Тэг	Тип	Описание	Частота (мс)
SysActiveUsers	A	Количество подключенных пользователей	60000
SysMinutesRun	A	Время (в минутах) с начала запуска	60000
SysPulse	D	Частота дискретизации (изменяется каждый час)	-
SysStatusEvent	E	Тэг состояния снимка события.	-
SysString	S	Системная строка (изменяется каждый час)	-

#### Переменные ошибок

Тэг	Тип	Описание	Частота (мс)
SysCritErrCnt	A	Количество критических ошибок с начала запуска	60000
SysErrErrCnt	A	Количество некритических ошибок с начала запуска	60000
SysErrorRate	A	Уровень системных ошибок.	60000
SysFatalErrCnt	A	Количество серьезных ошибок с начала запуска	60000

SysWarnErrCnt	A	Количество предупреждений с начала запуска	60000
---------------	---	--	-------

**Переменные даты**

Тэг	Тип	Описание	Частота (мс)
SysDateDay	A	Системная дата: число	300000
SysDateMonth	A	Системная дата: месяц	300000
SysDateYear	A	Системная дата: год	300000

**Переменные времени**

Тэг	Тип	Описание	Частота (мс)
SysTimeHour	A	Системное время: часы	300000
SysTimeMin	A	Системное время: минуты	60000
SysTimeSec	A	Системное время: секунды	1000

**Переменные места на диске**

Тэг	Тип	Описание	Частота (мс)
SysSpaceAlt	A	Осталось места для дополнительных данных - (100 Кбайт)	300000
SysSpaceBuffer	A	Осталось места для данных буфера - (100 Кбайт)	300000
SysSpaceMain	A	Осталось места для циклических данных - (100 Кбайт)	300000
SysSpaceWorm	A	Осталось места для постоянных данных - (100 Кбайт)	300000

**Статистика ввода-вывода**

Тэг	Тип	Описание	Частота (мс)
SysIODriverBytes	A	Количество байт от драйвера ввода-вывода (IODRIVER.EXE) с начала запуска.	60000
SysIODriverCPS	A	Символов в секунду от драйвера ввода-вывода (IODRIVER.EXE).	60000
SysStatusBytes	A	Количество байт от внутреннего системного драйвера (SYSDRV.EXE) с начала запуска.	60000
SysStatusCPS	A	Символов в секунду от внутреннего системного драйвера (SYSDRV.EXE).	60000

## Переменные статуса

Тэг	Тип	Описание	Значения
SysAnalogStorage	D	Статус хранения аналоговых значений	0 = Плохо 1 = Хорошо
SysDatabaseServer	D	Статус сервера баз данных	0 = Плохо 1 = Хорошо
SysDataReception	D	Прием данных от драйвера ввода-вывода	0 = Плохо 1 = Хорошо
SysDiscreteStorage	D	Статус хранения дискретных значений	0 = Плохо 1 = Хорошо
SysEventSystem	D	Статус системного события	0 = Плохо 1 = Хорошо
SysHealth	D	Целостность системы	0 = Плохо 1 = Хорошо
SysInSQLIOS	D	Статус сервера ввода-вывода InSQL	0 = Плохо 1 = Хорошо
SysIODriver	D	Статус драйвера ввода-вывода	0 = Плохо 1 = Хорошо
SysMisc	D	Статус хранения прочих значений	0 = Плохо 1 = Хорошо
SysRealTimeServer	D	Статус сервера реального времени (Realtime RT)	0 = Плохо 1 = Хорошо
SysStringStorage	D	Статус хранения текстовых значений	0 = Плохо 1 = Хорошо

## Переменные событий

Тэг	Тип	Описание	Значения	Частота (мс)
SysEventActionThreads	A	Количество потоков действий	>0	60000
SysEventDetectorThreads	A	Количество потоков детекторов	>0	60000
SysEventCritActionQSize	A	Размер очереди критических действий	>0	60000
SysEventNormActionQSize	A	Размер очереди нормальных действий	>0	60000

Ниже приводится пример запроса с использованием переменных целостности.

## Определение целостности системы

Следующий запрос является примером определения целостности одного узла системы IndustrialSQL Server. Этот запрос возвращает значения переменных целостности в течение последних двух минут с интервалом в одну секунду.

```
SET ww_Resolution 1000

SELECT SysMinutesRun, SysTimeMin, SysTimeSec
FROM AnalogWideHistory
WHERE DateTime > DateAdd(mi, -2, GetDate())
AND DateTime < GetDate()
```

Данные анализатора запросов (SQL Server Query Analyzer) будут выведены, например, в таком виде:

SysMinutesRun	SysTimeMin	SysTimeSec
24220.0	6.0	54.0
24220.0	6.0	55.0
24220.0	6.0	56.0
24220.0	6.0	57.0
24220.0	6.0	58.0
24220.0	6.0	59.0
24221.0	7.0	0.0
24221.0	7.0	1.0
24221.0	7.0	2.0
.		
.		
24222.0	8.0	54.0

(120 row(s) affected)

Значения в столбце "SysTimeSec" увеличиваются на 1 в каждой следующей строке, а значения в столбцах "SysMinutesRun" и "SysTimeMin" должны нарастать дважды.

## Производительность системы

Программа для оценки производительности системы Microsoft NT Performance Monitor (PERFMON.EXE) позволяет отслеживать системные переменные, описывающие работу операционной системы и оборудования. Программа Performance Monitor входит в комплект как Windows NT Workstation, так и Windows NT Server и может использоваться для наблюдения за удаленным компьютером с операционной системой Windows NT.

В этом разделе даются некоторые советы по использованию программы Performance Monitor для эффективного наблюдения за IndustrialSQL Server.

➤ **Как запустить программу Performance Monitor:**

1. В меню **Start** на панели задач Windows выберите **Programs**, затем **Administrative Tools**.
2. Выберите **Performance Monitor**.
3. Если программа не была предварительно настроена, то после запуска главное окно будет пустым.

## Счетчики

Программа Performance Monitor позволяет просматривать различные виды счетчиков, имеющихся в операционной системе Windows NT. Счетчики привязываются к объектам и источникам. Объектами могут быть память, процессы, серверы, система и т.д.. Источниками объектов могут быть, например, конкретные процессы. Можно выбрать один или несколько источников процессов для наблюдения за показаниями счетчиков, например, все процессы IndustrialSQL Server, выполняемые в среде Windows NT.

После того, как выбраны объекты и источники, выбираются соответствующие счетчики. Можно отслеживать такие показатели, как процент загрузки процессора, объем выделенной и свободной памяти и т.п. Набор счетчиков зависит от выбранных объектов и источников.

Можно также обеспечить детальный просмотр доступных счетчиков.

## Режимы просмотра

Программа Performance Monitor может отображать данные в четырех режимах: Chart (диаграмма), Report (Отчет), Alert (Сигнал) и Log (Журнал).

Диаграмма (Chart View) показывает состояние выбранных счетчиков, таких, как Загрузка процессора (%Processor Time), Загрузка диска (%Disk Time) и т.д. Этот режим позволяет выводить показания за определенный период времени. Отчет (Report View) выводит данные в виде таблицы.

Режим Сигнал (Alert View) используется для наблюдения за сервером и генерации сигналов (событий) при возникновении определенных условий. По сигналу можно запустить программу, сделать запись в журнале событий, отправить сообщение на другой компьютер и т.п.

С помощью режима Журнал (Log View) можно создавать файлы журналов для разных объектов. Затем эти данные могут быть проанализированы с помощью другой программы или импортированы в другое приложение, например в Excel.

Дополнительную информацию об использовании режимов просмотра счетчиков в программе Performance Monitor см. в *Руководстве администратора системы FactorySuite*.

## Загрузка сервера

Когда пользователь подключается к серверу IndustrialSQL Server с помощью программы-клиента (например, InSQL Trend), с сервера IndustrialSQL немедленно запрашиваются сведения о конфигурации. Эти данные включают в себя тэги, которые хранятся на сервере, их описания и т.п.. SQL-сервер считывает эту информацию из базы данных (записанной на диске) и размещает ее в памяти.

Например, когда пользователь выбирает периоды тренда, сервер IndustrialSQL считывает данные из файлов на диске и подготавливает результаты обработки запроса для отправки клиенту. Способность сервера быстро обрабатывать последовательность запросов от одного или нескольких клиентов зависит от максимального объема данных, которые можно хранить в памяти без повторного считывания этих данных с диска.

После загрузки системы ввода-вывода в память сервер должен использовать кэширование диска и считывать данные из кэша.

Ниже приводится таблица загрузки ресурсов различными подсистемами.

<b>Система</b>	<b>Описание загрузки</b>
Обмен и хранение данных	Основная загрузка IndustrialSQL Server. Происходит в течение работы системы. Не отражается на работе клиентов.
Считывание	Загрузка данных происходит по мере их выборки клиентскими приложениями. При подключении клиента запрашиваются данные конфигурации, которые хранятся на SQL Server. Программа IndustrialSQL Server запрашивает данные с SQL Server и загружает их.  Когда клиент запрашивает архивные данные, эта информация пересылается в память. Это происходит и при выборке дополнительной информации. Если клиентское приложение запрашивает данные, которые уже пересланы в память, операции с диском уже не производятся.

Сервер должен корректно обрабатывать различные ситуации при работе с клиентскими приложениями. Для этого необходимо обеспечить достаточные ресурсы для подсистем выборки и хранения данных.

## Журналы событий Windows NT

В операционной системе Windows NT ведется учет внутренних "событий". Событиями могут быть сообщения об ошибках, вызванные действиями системы или сгенерированными программами.

Система управления событиями Windows NT ведет три журнала: журнал безопасности (Security log), журнал приложений (Application log) и системный журнал (System log). Эти журналы содержат информацию обо всех неполадках в работе системы. Их необходимо регулярно проверять, чтобы убедиться, что система работает правильно.

Дополнительную информацию о журналах событий Windows NT см. в *Руководстве администратора системы FactorySuite*.

## Просмотр событий Windows NT

Программа просмотра событий (Event Viewer) является административным средством Windows NT для работы с журналами. Она позволяет вести журналы на рабочих станциях или серверах NT. Эту программу следует использовать для проверки следующих типов сообщений:

- Сообщения об ошибках программ IndustrialSQL
- Сообщения об ошибках операционной системы
- Подтверждения (выполнения запланированного события)

В зависимости от настройки и состояния системы журналы могут содержать сотни сообщений. Важно знать, что означают эти сообщения и какие действия нужно выполнить.

Дополнительную информацию о просмотре событий Windows NT см. в *Руководстве администратора системы FactorySuite*.

Чтобы использовать программу Windows NT Event Viewer для просмотра журналов ошибок IndustrialSQL Server, необходимо правильно настроить соответствующие значения реестра для IndustrialSQL Server.

[Дополнительную информацию о настройках журналов ошибок IndustrialSQL см. в главе 14, "Журналы ошибок"](#).

# Резервное копирование данных

В данной главе рассматриваются вопросы создания резервных копий баз данных IndustrialSQL Server и архивных файлов. Здесь описывается:

- Как выполнить копирование архивного блока.
- Как создать резервные копии баз данных IndustrialSQL Server.
- Как восстановить базы данных IndustrialSQL Server.

Для резервного копирования архивных данных используется утилита Backup Microsoft Windows NT. Для резервного копирования баз данных используется программа Microsoft SQL Server Enterprise Manager. В этой главе содержатся только инструкции по выполнению простого копирования и восстановления. Полную информацию об использовании этих приложений см. в документации Microsoft.

## Содержание

- [Копирование и восстановление архивных данных](#)
- [Копирование архивных блоков](#)
- [Копирование и восстановление баз данных](#)

## Копирование и восстановление архивных данных

Архивные данные хранятся на диске в формате файла блоков. Для повышения производительности при работе с реляционными базами данных сервер хранит большие порции данных на жестком диске, обеспечивая высокоскоростной доступ к этим данным из сети. После того, как созданы блоки архивных данных, может потребоваться перемещение этих данных на носитель длительного хранения, чтобы высвободить пространство для новых архивных блоков .

В качестве носителя для длительного хранения используется магнитная лента. Все операции по резервному копированию выполняются с помощью утилиты Windows NT Backup (NTBACKUP.EXE).

Подробную информацию о том, как копировать и восстанавливать файлы данных с помощью утилиты Windows NT Backup, см. в *Руководстве администратора системы FactorySuite*.

---

**Внимание!** Для резервного копирования необходимо использовать соединение с Microsoft SQL Server. Это соединение обычно определяется именем компьютера, на котором установлен Microsoft SQL Server.

---

Несмотря на то, что архивные данные , хранящиеся на магнитной ленте, не могут быть считаны клиентскими приложениями непосредственно с ленты, их можно скопировать на жесткий диск. Если вам известна текущая запись на магнитной ленте и содержащиеся в ней файлы, то задача выборки информации упрощается.

## Копирование архивных блоков

Архивные блоки или их подмножество можно скопировать в "архивную" область (постоянную область хранения). Это позволяет делать "снимок" событий в течение промежутка времени, размер которого зависит от длины архивного блока. Например, длина промежутка составляет 2 часа. Эти 2 часа будут помещены в блок архивных данных, длина которого обычно позволяет хранить данные, полученные в течение 24 часов. "Снимок" этих 2 часов можно сохранить в архиве.

---

**Внимание!** Для выполнения резервного копирования необходимо использовать соединение с Microsoft SQL Server. Это соединение обычно определяется именем компьютера, на котором установлен Microsoft SQL Server.

---

При копировании архивных блоков должны выполняться следующие условия:

- Путь *<IndustrialSQL Server>* должен быть указан в переменных среды.
- Модуль XPPDS.DLL должен находиться в каталоге *<путь IndustrialSQL Server>*
- Должна быть установлена хранимая процедура **xp\_DiskCopy**. (Обычно она устанавливается вместе с IndustrialSQL Server.)

Для копирования подмножества данных используйте расширенную процедуру копирования **xp\_DiskCopy**. Синтаксис этой процедуры следующий:

```
xp_DiskCopy StartTime, EndTime, Description
```

Дата/время должны иметь формат: MM/DD/YY или MM/DD/YYYY  
HH:MM:SS.MS. Например:

```
xp_DiskCopy '02/20/1999 13:10:00', '02/20/1999  
14:00:00', 'Plant trip 02/20/1999'
```

---

**Примечание.** Microsoft SQL Server кэширует ранее полученные данные для последующего использования в текущем сеансе работы. Если архивные данные считываются каким-либо другим способом, клиенты должны отключиться и заново подключиться к серверу для того, чтобы данные кэша были обновлены.

---

Расширенные процедуры копирования могут использоваться только при подключении к "главной" базе данных. Для получения справки используйте процедуру **xp\_proclist**.

Дополнительную информацию о расширенных процедурах копирования см. в *Справочном руководстве IndustrialSQL Server*.

## Копирование и восстановление баз данных

Настоятельно рекомендуется создавать резервные копии баз данных IndustrialSQL Server и SQL в следующих случаях:

- Перед внесением любых изменений в базы данных.
- По установленному графику, чтобы минимизировать риск потери данных в случае сбоя диска. Лучшим решением будет использование автоматического копирования с помощью программы SQL Server Enterprise Manager. Необходимо создавать резервные копии баз данных не реже одного раза в неделю.

При выполнении резервного копирования все системные таблицы, объекты и данные пользователя копируются на отдельное устройство, называемое носителем копий или дампов. Этими устройствами могут быть файлы, дискеты, zip-диски и магнитные ленты. Процессом резервного копирования легко управлять с помощью программы SQL Server Enterprise Manager.

В этом разделе описывается процесс простого копирования и восстановления базы данных. Полное описание копирования и восстановления баз данных с помощью Microsoft SQL Server Enterprise Manager, включая рекомендации по планированию и ведению журналов, см. в документации Microsoft.

## Резервное копирование базы данных

Резервное копирование баз данных необходимо выполнять в соответствии с установленным графиком и перед внесением изменений в базу данных. В этом разделе описывается, как выполнять резервное копирование в существующий файл и, как создать новую копию. Файл резервной копии также называется резервным устройством. Резервное копирование следует выполнять с помощью соединения с Microsoft SQL Server; в противном случае копия будет пуста. Соединение с Microsoft SQL Server обычно определяется именем компьютера, а котором установлен Microsoft SQL Server.

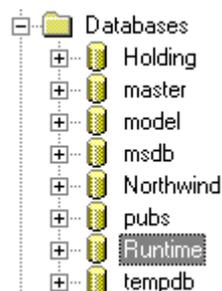
---

**Примечание.** Все выполняемые процессы при выполнении резервного копирования будут свернуты.

---

### ➤ Как создать резервную копию базы данных:

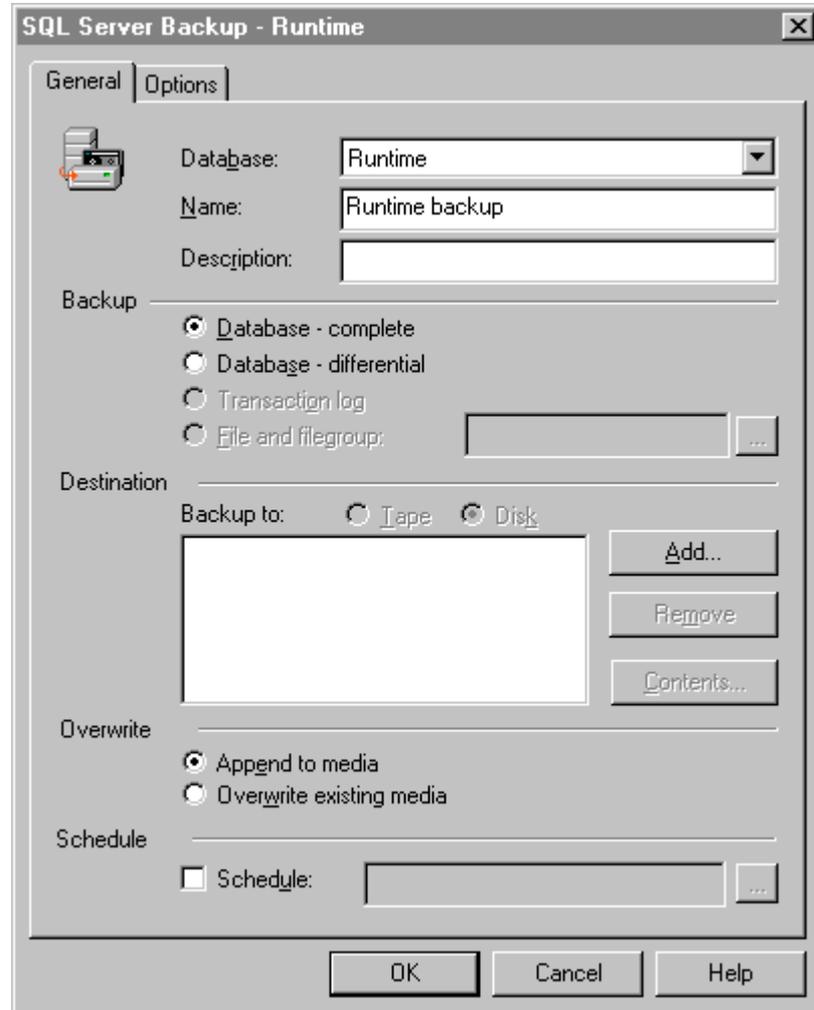
1. Запустите программу SQL Server Enterprise Manager.
2. Выберите базу данных, которую нужно скопировать. Например, *Runtime*.



3. Нажмите правой кнопкой мыши на имени базы данных. В контекстном меню выберите **All Tasks**, затем **Backup Database**.

Появится диалоговое окно **SQL Server Backup**.

4. Откройте вкладку **General**.



5. В группе **Database** выберите базу данных для копирования.
6. Чтобы использовать существующий файл копии, выберите нужное устройство в группе **Destination**. Перейдите к шагу 13.

---

**Совет.** Для получения подробных сведений об устройстве выберите устройство из списка и нажмите **Contents**.

---

7. Чтобы создать новую копию, нажмите **Add**.

Появится диалоговое окно **Choose Backup Destination**.



8. В поле **File name** введите путь для новой резервной копии. Убедитесь в наличии свободного места для сохранения резервной копии.
9. Из списка **Backup device** выберите <New Backup Device>.

Появится диалоговое окно **Backup Device Properties**.



10. В поле **Name** введите имя резервной копии.  
После ввода имени будет изменен путь для резервной копии. Проверьте его правильность.
11. Для создания нового файла резервной копии нажмите **OK**.
12. Нажмите **OK**, чтобы закрыть диалоговое окно **Choose Backup Destination**.  
Теперь созданное устройство появится в группе **Destination** диалогового окна **SQL Server Backup**. Выберите новое устройство резервного копирования.
13. Нажмите **OK** для выполнения копирования.

Вы можете настроить различные параметры резервного копирования, например, срок действия резервной копии, а также запланировать автоматическое резервное копирование.

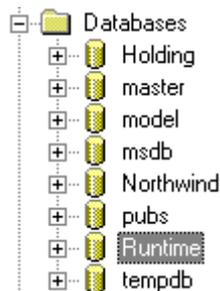
Дополнительную информацию о резервном копировании баз данных с помощью программы SQL Server Enterprise Manager см. в документации Microsoft.

## Восстановление базы данных

При восстановлении базы данных с резервной копии вся информация, хранящаяся в базе, будет перезаписана. Все изменения, внесенные после выполнения резервного копирования, будут потеряны. Кроме того, все выполняемые процессы будут свернуты.

### ➤ Как восстановить базу данных:

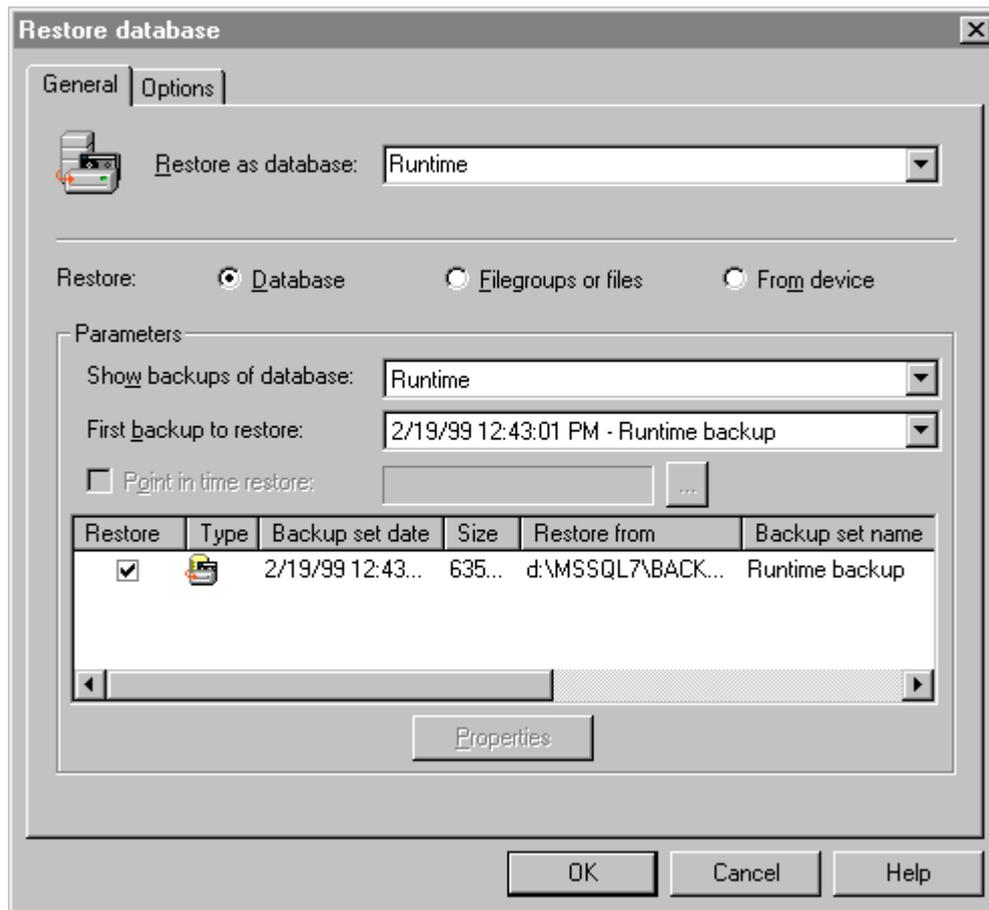
1. Запустите программу SQL Server Enterprise Manager.
2. Выберите базу данных, которую нужно восстановить. Например, *Runtime*.



3. Нажмите правой кнопкой мыши на имени базы данных. В контекстном меню выберите **All Tasks**, затем **Restore Database**.

Появится диалоговое окно **Restore Database**.

4. Откройте вкладку **General**.



5. В списке **Restore as database** выберите базу данных для восстановления.
6. Пометьте параметр **Database** в группе **Restore**.
7. В поле **First backup to restore list** выберите нужную копию.
8. Нажмите **OK**. Данные будут восстановлены.

Вы можете настроить различные параметры восстановления баз данных. Дополнительную информацию о резервном копировании баз данных с помощью программы SQL Server Enterprise Manager см. в документации Microsoft.

# Настройка серверов ввода-вывода

В этой главе рассказывается, как просматривать, редактировать и удалять существующие серверы ввода-вывода, а также как создавать и настраивать новые серверы ввода-вывода.

Подробнее о просмотре ошибок процессов ввода-вывода (IODRIVER.EXE) см. в главе 14, "Сообщения об ошибках".

## Содержание

- О серверах ввода-вывода
- Настройка серверов ввода-вывода через InTouch
- Настройка серверов ввода-вывода через InSQL Configure
- Синхронизация метки времени и хранение данных
- Сервер ввода-вывода InSQL
- Настройки реестра для драйверов устройств

## О серверах ввода-вывода

Сервером ввода-вывода называется приложение, поставляющее данные клиенту через определенный протокол. Существует три типа серверов ввода-вывода, используемых в IndustrialSQL Server:

- Внешние серверы ввода-вывода, которые являются приложениями, поставляющими данные в IndustrialSQL Server через протокол DDE или SuiteLink. Примерами являются серверы ввода-вывода Wonderware, серверы ввода-вывода сторонних фирм и созданные пользователями.
- Системный драйвер ввода-вывода, который является внутренним драйвером.
- Сервер ввода-вывода InSQL, который является интерфейсом клиентов для обращения к данным реального времени IndustrialSQL Server.

Термины "сервер" и "клиент" — родственные. Для передачи данных с производственного участка в приложение-клиент, такое как InSQL Trend, данные должны пройти обработку в цепи приложений. Любое приложение, получающее данные из нижестоящего источника, есть клиент. Любое приложение, посылающее данные в вышестоящий источник, называется сервером. IndustrialSQL Server является и клиентом, и сервером; это клиент для серверов ввода-вывода и сервер для клиентов на рабочих станциях (таких как InSQL Trend).

Все серверы ввода-вывода компании Wonderware имеют адресную информацию, составленную из трех частей: имени приложения (включая имя компьютера, на котором запущен сервер), темы и элемента данных.

Именем приложения называется имя поставляющего данные приложения. Примером имени приложения (включая имя сервера) может служить: `\\FactoryServer\Excel`. Это имя повлечет обращение к программе Excel, запущенной на сервере ввода-вывода "FactoryServer".

Тема — это определяемая приложением подгруппа элементов данных. Примерами тем могут служить имена электронных таблиц Excel или "Имя тэга" для InTouch WindowViewer. Элементом называется заглушка значения данных. Примером элемента служит имя ячейки в электронной таблице Excel. Другим примером может служить имя тэга в InTouch.

Существует два способа настройки серверов ввода-вывода: импортировать информацию о существующем сервере ввода-вывода из InTouch или же добавить ее с помощью InSQL Configure.

## Настройка серверов ввода-вывода через InTouch

Можно настроить сервер ввода-вывода, определив сервер ввода-вывода в InTouch и затем импортировав базу данных тэгов (через файлы TAGNAME.X или DDE.CFG) в IndustrialSQL Server при помощи функции импорта программы InSQL Configure. Если IndustrialSQL Server интегрирован с существующей системой InTouch, то данный способ является самым быстрым для внесения всей информации настройки сервера ввода-вывода.

После импортирования базы данных тэгов становится возможным просмотр, редактирование или удаление серверов ввода-вывода и тем в IndustrialSQL Server.

Подробнее об импортировании базы данных тэгов InTouch см. в главе 3, "Интеграция InTouch".

---

**Примечание.** Можно импортировать словари тэгов из InTouch версии 6.0 и выше. Однако для дельта-импортирования следует использовать базу данных тэгов InTouch 7.1.

---

## Настройка серверов ввода-вывода через InSQL Configure

В программе InSQL Configure при выборе папки **I/O Servers** в панели браузера будет показан список настроенных серверов ввода-вывода. Панель свойств серверов ввода-вывода позволяет просматривать, редактировать и удалять существующие серверы ввода-вывода и связанные с ними темы. Также можно добавлять новые серверы ввода-вывода и темы. Невозможно создать тэг, если недоступны сервер ввода-вывода и соответствующая тема. [Подробнее о запуске и использовании InSQL Configure см. в главе 2, "InSQL Configure"](#).

### Редактирование информации сервера ввода-вывода

После редактирования информации сервера ввода-вывода и последующего повторного импортирования базы данных тэгов при помощи InSQL Configure изменения сервера ввода-вывода не будут сохранены.

➤ **Как отредактировать информацию сервера ввода-вывода:**

1. В панели групп InSQL Configure выберите сервер ввода-вывода, который нужно изменить.
2. Разверните папку **I/O Servers**.
3. Выберите имя сервера ввода-вывода для редактирования.

В панели **General** появится информация о сервере ввода-вывода.

The screenshot shows the 'General' configuration window for an I/O server. It contains the following fields and options:

- Computer Name:** A text input field containing the value '123238'.
- I/O Server Type:** A dropdown menu with 'modbus' selected.
- Description:** A text input field containing the value 'Modicon Modbus'.
- Protocol Type:** A group box containing two radio buttons: 'DDE' (unselected) and 'SuiteLink' (selected).
- Redirect to InTouch:** A button located at the bottom of the window.

4. В поле **Description** введите описание сервера ввода-вывода.
5. В списке **I/O Server Type** выберите тип сервера. Например, если данные поступают из InTouch WindowViewer, следует выбрать пункт "View".  
**О добавлении сервера ввода-вывода см. раздел "Добавление сервера ввода-вывода" далее в этой главе.**
6. В поле **Computer Name** выберите имя компьютера, на котором запущен сервер ввода-вывода.
7. Выберите протокол, который будет использовать сервер ввода-вывода для передачи данных в IndustrialSQL Server.

#### **DDE**

DDE есть протокол передачи данных между приложениями без участия пользователя и без просмотра. В среде Windows передача DDE сжимается через набор сообщений, процедур (протоколов) обработки этих сообщений и некоторых новых типов данных. При использовании этих протоколов приложения, написанные независимо друг от друга, могут передавать данные друг другу без участия пользователя. Например, InTouch и Excel.

#### **SuiteLink**

SuiteLink — это новый протокол, предоставляющий приложениям большую функциональность, чем в случае с DDE. Протокол SuiteLink обеспечивает передачу меток времени и качества в пакетах данных.

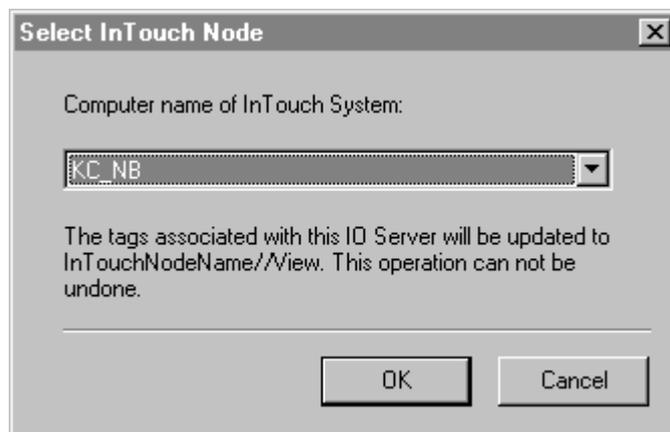
#### **System**

Системный протокол является внутренним средством передачи данных из системных переменных в IndustrialSQL Server.

8. Для перенаправления сервера ввода-вывода в InTouch выберите **Redirect to InTouch**.

При перенаправлении сервера ввода-вывода в InTouch вы задаете получение значений тэгов через конкретный узел InTouch, использующий сервер ввода-вывода, вместо получения их непосредственно с сервера ввода-вывода. Это полезно, если требуется уменьшить загрузку ресурсов сервера ввода-вывода, или если узел InTouch более доступен в сети.

Появится диалоговое окно **Select InTouch Node**.



9. Выберите из списка имя узла InTouch, с которого вы хотите получать значения тэгов.

При перенаправлении сервера ввода-вывода имя компьютера и тип сервера ввода-вывода будут отражать узел InTouch как сервер ввода-вывода, с которого принимаются данные. Например, предположим, что используется сервер ввода-вывода Modicon Modbus на компьютере "I23238". Текущая тема покажет `\\I23238\modbus`. При направлении этого сервера ввода-вывода на узел InTouch "InTouchNode1" тема изменится на `\\InTouchNode1\view`.

Если было импортировано несколько узлов InTouch, убедитесь в том, что выбранный узел InTouch получает данные с того сервера ввода-вывода, который вы перенаправляете.

---

**Внимание!** После перенаправления сервера ввода-вывода (нажата кнопка **Save**) отменить эти действия уже нельзя.

---

10. Нажмите **OK** для перенаправления сервера ввода-вывода.
11. Для сохранения результатов нажмите кнопку . Или же в меню **File** выберите **Save**.
12. Для отмены изменений нажмите кнопку . Или же в меню **File** выберите **Undo**.

Подробнее о протоколах DDE и SuiteLink см. в *Руководстве администратора системы FactorySuite*.

## Добавление сервера ввода-вывода

При добавлении нового сервера ввода-вывода в таблицу *IOServer* базы данных *Runtime* добавляется новая строка.

### ➤ Как добавить сервер ввода-вывода:

1. В меню **File** выберите **New**. Или нажмите кнопку  на панели инструментов.

Появится диалоговое окно **New**.



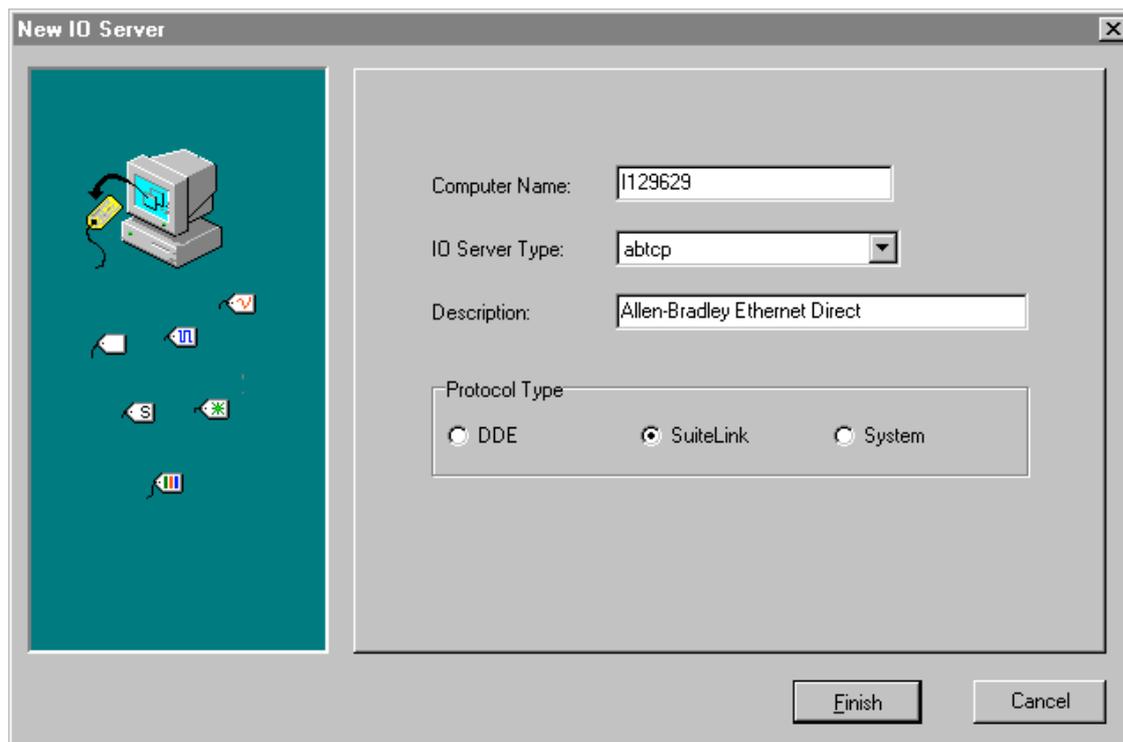
---

**Совет.** Нажмите в этом окне правой кнопкой мыши для выбора режима отображения ярлыков.

---

2. Выберите ярлык **IO Server** и нажмите **OK**.

Появится диалоговое окно **New IO Server**.



3. Введите информацию настройки нового сервера ввода-вывода. Подробнее об этих параметрах см. в разделе "Редактирование информации сервера ввода-вывода" ранее в этой главе.
4. Завершив настройку нового сервера ввода-вывода, нажмите **Готово**.

## Удаление сервера ввода-вывода

После удаления сервера ввода-вывода и последующего повторного импортирования базы данных тэгов, содержащей определения сервера ввода-вывода, при помощи InSQL Configure этот сервер ввода-вывода будет снова добавлен.

Сервер ввода-вывода не может быть удален, если он все еще имеет связанные с ним темы.

### ➤ Как удалить сервер ввода-вывода:

1. В панели групп браузера InSQL Configure выберите сервер ввода-вывода, который нужно удалить.
2. Разверните папку **I/O Servers**.
3. Удалите сервер ввода-вывода любым из следующих способов:
  - Нажмите правую кнопку мыши на сервере ввода-вывода. В контекстном меню выберите **Delete**.
  - Выберите сервер ввода-вывода и нажмите кнопку  на панели инструментов.

## Редактирование информации тем

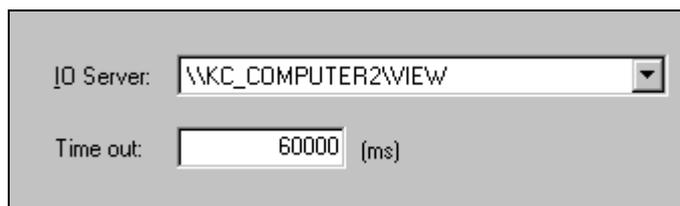
Тема — это логический блок данных сервера ввода-вывода. Протоколы DDE и SuiteLink используют темы для идентификации информации, поступающей с сервера ввода-вывода.

После редактирования информации сервера ввода-вывода и последующего повторного импортирования базы данных тэгов при помощи InSQL Configure изменения сервера ввода-вывода не будут сохранены.

### ➤ Как отредактировать информацию тем:

1. В панели групп броузера InSQL Configure выберите папку серверов ввода-вывода для показа всех существующих в системе серверов ввода-вывода.
2. Выберите сервер ввода-вывода, содержащий тему, которую нужно редактировать.
3. В соответствующей панели выберите нужную тему.

Панель свойств отобразит информацию для выбранной темы.



4. В списке **IO Server** выберите сервер ввода-вывода, содержащий нужную тему.  
[Подробнее о создании сервера ввода-вывода см. в разделе "Добавление сервера ввода-вывода" ранее в этой главе.](#)
5. В поле **Time Out** введите временной интервал в миллисекундах, за который должны быть получены данные этой темы.

Если за этот промежуток времени данные не получены, тема считается "мертвой" и все тэги этой темы получают метку "плохое качество", которая останется с ними до тех пор, пока не будет получено новое значение тэга.

6. Для сохранения результатов нажмите кнопку . Или же в меню **File** выберите **Save**.
7. Для отмены изменений нажмите кнопку . Или же в меню **File** выберите **Undo**.

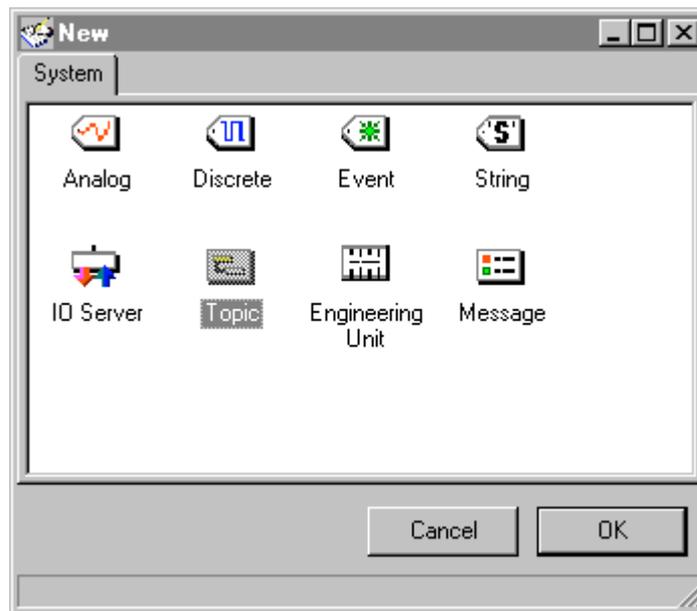
## Добавление темы

При добавлении новой темы для сервера ввода-вывода в таблицу *Topic* базы данных *Runtime* добавляется новая строка.

### ➤ Как добавить тему:

1. В меню **File** выберите **New**. Или нажмите кнопку  на панели инструментов.

Появится диалоговое окно **New**.



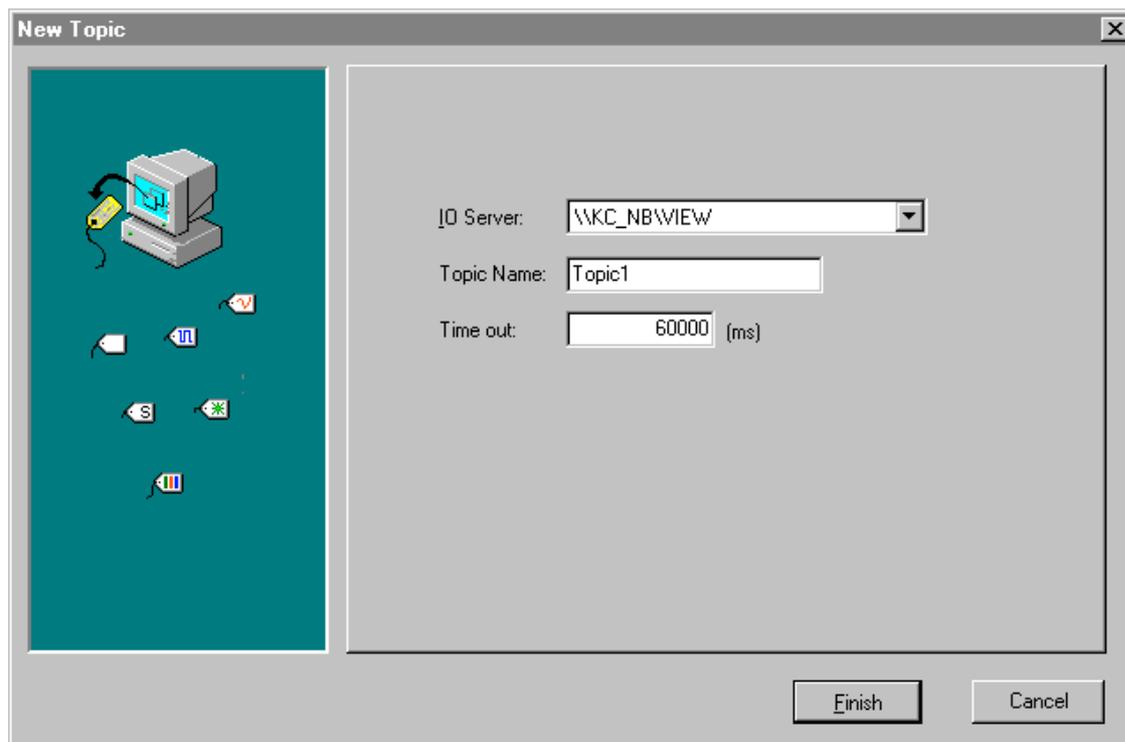
---

**Совет.** Нажмите в этом окне правой кнопкой мыши для выбора режима отображения ярлыков.

---

2. Выберите ярлык **Topic** и нажмите **OK**.

Появится диалоговое окно **New Topic**.



3. В поле **Topic Name** укажите уникальное имя темы.

Имена тем должны быть уникальны для сервера ввода-вывода, а не для всей системы в целом. Две темы с идентичными именами могут существовать до тех пор, пока они относятся к разным серверам ввода-вывода.

4. Введите информацию настройки для новой темы.  
[Подробнее об этих параметрах см. в разделе "Редактирование информации тем" ранее в этой главе.](#)
5. Закончив создание новой темы, нажмите **Finish**.

## Удаление темы

После удаления темы и последующего повторного импортирования базы данных тэгов, содержащей определения сервера ввода-вывода, при помощи InSQL Configure эта тема будет снова добавлена в базу данных. Тема не может быть удалена, пока она имеет связанные с ней тэги.

### ➤ Как удалить тему:

1. В панели групп браузера InSQL Configure выберите сервер ввода-вывода, в котором нужно удалить тему.
2. Разверните папку **I/O Servers**.
3. Дважды нажмите левую кнопку мыши на сервере ввода-вывода, содержащем тему, которую требуется удалить.
4. Удалите тему одним из следующих способов:
  - Нажмите правую кнопку мыши на теме. В появившемся меню выберите **Delete**.

- Выберите тему и нажмите кнопку  на панели инструментов.

## Синхронизация метки времени и хранение данных

Все серверы ввода-вывода, поддерживающие SuiteLink, создают метку времени для производственных данных при их получении. Далее описывается, как происходит синхронизация между меткой времени, приходящей от сервера ввода-вывода, и текущим временем IndustrialSQL Server.

### Изменение метки времени на сервере ввода-вывода

Для синхронизации метки времени сервера ввода-вывода и времени IndustrialSQL Server систему IndustrialSQL Server можно настроить так, чтобы она отсылала сообщения в серверы ввода-вывода с инструкцией изменить время в их локальных метках. По умолчанию это сообщение отсылается каждые 60 минут. Интервал отправки сообщения о синхронизации метки времени определяется значением *TimeSyncInterval* в следующем ключе реестра:

```
HKEY_LOCAL_MACHINE\SOFTWARE
  \Wonderware
    \InSQL
      \Programs
        \Comms
          \2
            \TimeSyncInterval
```

Если значение *TimeSyncInterval* установлено на 0, то синхронизации времени не происходит.

При получении этого сообщения сервер ввода-вывода изменяет метку времени для данных, после чего она совпадает с текущим временем в IndustrialSQL Server. Сервер ввода-вывода не изменяет свои локальные часы.

Синхронизация не применяется для серверов ввода-вывода, использующих DDE, так как на этих серверах метка времени не ставится. Для данных, приходящих с серверов ввода-вывода через DDE, всегда используется время IndustrialSQL Server.

Если тема отсоединилась/воссоединилась из-за превышения тайм-аута темы или другой ошибки связи, время сервера ввода-вывода не будет обновлено до истечения интервала синхронизации.

## Изменение метки времени в IndustrialSQL Server

Далее описано, как IndustrialSQL Server создает измененные метки времени. Industrial SQL Server совершает это действие между синхронизациями меток времени серверов ввода-вывода, когда метка времени нарушает синхронность (сдвигается вперед или назад) по отношению ко времени Industrial SQL Server.

### Время сервера ввода-вывода отстает

Если метка времени сервера ввода-вывода отстает от времени IndustrialSQL Server менее, чем на 30 секунд, никаких изменений с этой меткой не происходит.

Если метка времени сервера ввода-вывода отстает от времени IndustrialSQL Server более, чем на 30 секунд, Industrial SQL Server создаст метку времени для данных равную своему времени, минус 30 секунд.

Получив от Industrial SQL Server сообщение о синхронизации, сервер ввода-вывода изменяет метку времени, что вызывает "пробел" в сохраненных данных. Этот "пробел" приходится на период от последней метки времени сервера ввода-вывода перед синхронизацией до первой метки, измененной сервером ввода-вывода.

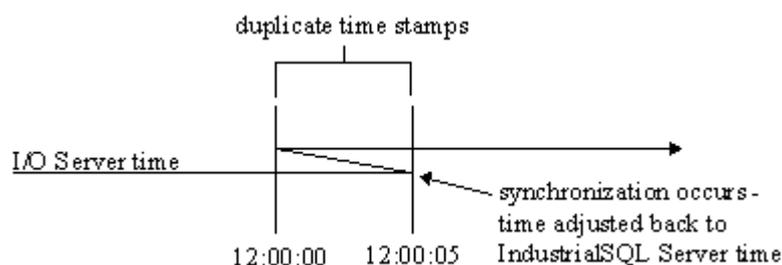
	I/O Server Time	IndustrialSQL Server Time
No data between	11:59:55	12:00:00
12:00:00 and 12:00:04	11:59:56	12:00:01
	11:59:57	12:00:01
	11:59:58	12:00:03
	11:59:59	12:00:04
synchronization occurs →	12:00:05	12:00:05
	12:00:06	12:00:06
	12:00:07	12:00:07
	12:00:08	12:00:08
	12:00:09	12:00:09
	12:00:10	12:00:10

## Метка времени убегает вперед

При циклическом сохранении метка времени сервера ввода-вывода может превысить (до 1 секунды) время IndustrialSQL Server без увеличения метки времени или качества. Если метка времени сервера ввода-вывода превышает время IndustrialSQL Server более, чем на 1 секунду, IndustrialSQL Server создаст метку равную времени IndustrialSQL Server плюс 1 секунда. Значение качества в этом случае будет сохранено как "неопределенное".

При дельта-сохранении локальное время сервера ввода-вывода может превысить (до 1 часа 10 минут) время IndustrialSQL Server без изменения метки времени или качества. Значение качества в этом случае будет сохранено как "неопределенное". Если время сервера ввода-вывода убегает вперед более, чем на 1 час и 10 минут, то данные не будут заноситься в архив пока время (а не метка времени) сервера ввода-вывода не будет синхронизировано с IndustrialSQL Server.

Если метка времени сервера ввода-вывода уменьшается при синхронизации, в системе могут появиться значения с одинаковой меткой времени. Например:



Для предотвращения перезаписи существующих данных метка времени делается равной времени последнего сохранения, плюс 5 миллисекунд. Качество изменяется на "неопределенное". Например:

	I/O Server Time	Stored Time Stamp
	12:00:00	12:00:00
	12:00:01	12:00:01
	12:00:02	12:00:02
	12:00:03	12:00:03
	12:00:04	12:00:04
Synchronization →	12:00:05	12:00:05
occurs	12:00:00	12:00:05:005
	12:00:01	12:00:05:010
	12:00:02	12:00:05:015
	12:00:03	12:00:05:020
	12:00:04	12:00:05:025
	12:00:05	12:00:05:030

Если чертить тренд с этими данными, значения с измененной меткой времени будут заключены в отрезок 1 секунды (в данном примере это 12:00:05) и не будут видны с разрешением в 1 секунду. Если увеличить масштаб этот временной отрезок, то можно заметить данные с измененной меткой времени.

## Работа системы с отключенной синхронизацией

Далее показано, как IndustrialSQL Server работает с метками времени сервера ввода-вывода при отключенной синхронизации. При запуске IndustrialSQL Server, если исходные метки времени данных сервера ввода-вывода не синхронизированы, IndustrialSQL Server автоматически изменит метку времени и качества данных для значений тэгов, приходящих с сервера ввода-вывода. Изменение зависит от того, сохраняется ли значение циклично или же методом исключения (дельта-хранение).

### Циклическое хранение

Если метка времени сервера ввода-вывода *отстает* от текущего времени IndustrialSQL Server более, чем на 30 секунд:

1. Метка времени будет изменена на ту же метку, что имеет IndustrialSQL Server, минус 30 секунд.
2. Атрибут *качества* для тэга примет значение "хорошее".
3. Атрибут *детального качества* примет значение "неопределенное".

Когда метка времени сервера ввода-вывода *опережает* текущее время IndustrialSQL Server более, чем на 1 секунду:

1. Метка времени будет изменена на ту же метку, что имеет IndustrialSQL Server, плюс 1 секунда.
2. Атрибут *качества* для тэга примет значение "хорошее".
3. Атрибут *детального качества* примет значение "неопределенное".

### Дельта-хранение

Если метка времени сервера ввода-вывода *отстает* от текущего времени IndustrialSQL Server, более, чем на 30 секунд:

1. Данные не сохраняются.
2. Будет создано сообщение об ошибке, когда система получит тысячное недопустимое значение времени.

Когда метка времени сервера ввода-вывода для тэга *опережает* текущее время IndustrialSQL Server более, чем на 3600 секунд (1 час):

1. Данные не сохраняются.
2. Будет создано сообщение об ошибке, когда система получит тысячное недопустимое значение времени.

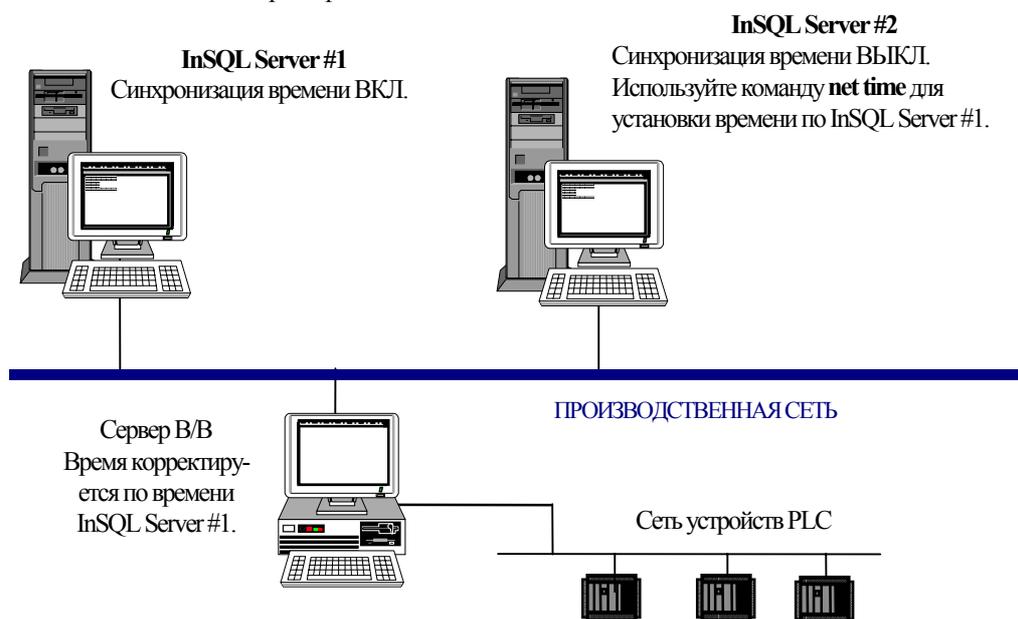
## Использование сервера ввода-вывода с несколькими системами IndustrialSQL Server

Если существует несколько систем IndustrialSQL Server, соединенных с одним сервером ввода-вывода, следует просто включить синхронизацию времени для одной системы IndustrialSQL Server и использовать этот IndustrialSQL Server для корректировки времени во всей структуре серверов.

Если синхронизация времени включена на всех серверах IndustrialSQL Server, а их установки времени различны, метка времени сервера ввода-вывода будет корректироваться по времени того сервера IndustrialSQL Server, от которого последним был получено сообщение синхронизации. Это может привести к возникновению неверных данных (дублированные метки времени или пробелы в данных).

Следует синхронизировать все системы IndustrialSQL Server и сервер ввода-вывода с одним IndustrialSQL Server хотя бы раз в день. Оптимальный вариант — синхронизировать один раз в час с каким-либо стабильным внешним источником, таким как универсальные часы.

Минимум, что можно сделать — это использовать команду **net time** (для операционной системы Windows NT) для установки времени других IndustrialSQL Server, основываясь на времени выбранного IndustrialSQL Server. Например:



Для выполнения команды **net time** откройте сеанс DOS на том компьютере, который нужно синхронизировать с выбранным IndustrialSQL Server, а затем наберите:

```
net time \\имякомпьютера /set
```

где *имякомпьютера* есть имя выбранного IndustrialSQL Server. (Добавьте параметр */у* для установки времени без подтверждения).

Синхронизацию времени можно также выполнить следующими способами:

1. Запишите команду синхронизации времени в пакетный файл, зациклив его на периодичность 1 раз в час.
2. Запланируйте команду синхронизации времени при помощи службы SQL Server Agent, доступной в SQL Server Enterprise Manager.
3. Используйте Windows NT Resource Kit Time Service для установки автоматической синхронизации времени.

Time Service работает как служба Windows NT. Можно настроить Time Service на автоматическую синхронизацию компьютеров с общим источником в указанный ежедневный интервал времени.

Подробнее см. документацию по Windows NT Resource Kit.

4. Используйте планировщик задач Windows NT для выполнения команды Windows "AT". (Удостоверьтесь в том, что эта служба включена).
5. Используйте программу синхронизации для корректировки времени по какому-либо внешнему источнику, такому как универсальные часы.

## Сервер ввода-вывода InSQL

Сервер ввода-вывода InSQL (INSQLIOS.EXE) — это интерфейс клиентов для получения данных из активного образа IndustrialSQL Server через протокол DDE или SuiteLink. Активный образ — это область памяти, в которой временно содержатся все значения полученных данных до их сохранения на диск. Сервер ввода-вывода InSQL может обновлять элементы на текущие значения для заданных тем, обеспечивая функциональность серверов ввода-вывода реального времени.

Например, сервер ввода-вывода InSQL может использоваться программой WindowViewer для обращения к системным тэгам IndustrialSQL Server, показывающим состояние целостности системы. Программу WindowViewer можно настроить на создание аларма при обнаружении ненормального поведения системы IndustrialSQL Server.

[Подробнее об активном образе см. в главе 4, "Управление системой хранения"](#).

По умолчанию сервер ввода-вывода InSQL работает как служба Windows NT, которую можно запустить и остановить программой InSQL Control. Из программы InSQL Control можно также просмотреть статус сервер ввода-вывода InSQL. Информация о статусе будет показана в главном окне InSQL Control.

[Подробнее о программе InSQL Control см. в главе 5, "InSQL Control"](#).

Сервер ввода-вывода InSQL является сервером только для чтения. Это значит, что InTouch (например) не может обновлять данные в активном образе.

## Темы сервера ввода-вывода InSQL

По умолчанию сервер ввода-вывода InSQL настроен на работу в качестве службы, использующей стандартный файл настройки для соединений клиентов. В протоколах DDE или SuiteLink тема — это определяемая приложением подгруппа элементов данных. Для идентификации тем в приложениях эти протоколы используют систему наименования из трех частей. Примером темы может служить имя электронной таблицы Excel.

---

**Примечание.** В этой версии сервер ввода-вывода InSQL заранее сконфигурирован с одной темой **Tagname**.

---

Сервер ввода-вывода InSQL будет принимать запросы клиентов (таких как WWClient или WindowViewer), делающих попытки установить соединение при использовании одной из заранее настроенных тем. После соединения клиента с сервером ввода-вывода InSQL между ними устанавливается "оперативная" связь.

Для успешного соединения с сервером ввода-вывода InSQL клиенты должны указывать в запросе допустимые темы. Допустимыми являются только темы, сконфигурированные на сервере ввода-вывода InSQL, поэтому любой клиент, указывающий тему, которой нет в сервере ввода-вывода InSQL, не получит соединения.

## Настройки реестра для драйверов устройств

В этом разделе описаны настройки, необходимые для добавления или удаления вручную драйверов устройств из системы IndustrialSQL Server.

---

**Примечание.** Сервер ввода-вывода является драйвером устройства.

---

Информация драйвера устройства находится под следующим ключом реестра Windows NT:

```
HKEY_LOCAL_MACHINE
  \SOFTWARE
    \Wonderware
      \InSQL
        \Programs
          \Comms
```

Можно настроить несколько драйверов для системы хранения IndustrialSQL Server посредством определения второстепенных ключей внутри ключа \Comms. Например, когда нужно настроить два драйвера, это будет выглядеть следующим образом:

```
\Comms
  \1
  \2
```

Количество второстепенных ключей указано в значении ключа \Comms, названном "Numprogs". В данном примере, значение "Numprogs" равно 2.

## Значение ключей настройки драйверов

Каждый ключ настройки драйвера содержит следующие простые значения. Значения REG\_SZ представлены одной текстовой строкой. Значения REG\_DWORD представлены цифрами.

```
\Comms
  \1
    Args: REG_SZ
    DeviceName: REG_SZ
    DeviceType: REG_DWORD
    GatewayName: REG_SZ
    Hidden: REG_DWORD
    Maximised: REG_DWORD
    Minimised: REG_DWORD
    Progname: REG_SZ
    SourceId: REG_DWORD
    Title: REG_SZ
```

Описания этих значений приведены в следующей таблице:

Значение	Описание
Args	Параметр командной строки, необходимый для запуска драйвера. Этот параметр идентифицирует второстепенный ключ, содержащий настройки драйвера. Значением является /CommId:1 в том случае, когда драйвер настроен из подключения \Comms \1. При настройке следующего драйвера значение аргумента второго драйвера будет /CommId:2.
DeviceName	Идентификатор устройства для драйвера, зависящего от устройства. Имеет смысл только, когда драйвер является модулем DLL. Имя устройства должно совпадать с второстепенным ключом, содержащим зависящую от устройства информацию о настройках.
DeviceType	Значение указателя из пронумерованного списка значений драйверов. Пронумерованный список находится в таблице <i>IOServer</i> базы данных IndustrialSQL Server.
GatewayName	В том случае, если данные устройства получаются через шлюзовой сервер, использующий сетевой драйвер протокола <i>named pipes</i> , это значение содержит имя шлюзового компьютера.
Hidden, Maximized, Minimized	Переключатели на основе булева выражения, определяющие режим запуска окна приложения.
Progname	Это значение содержит полный путь и имя основного файла-исполнителя драйвера устройства. Когда зависящий от устройства код выполнен в виде DLL, это значение должно быть именем файла-исполнителя, связанного с DLL при выполнении.
SourceId	Последовательный номер источника драйверов.
Title	Текстовое значение, используемое для описания драйвера.

Подробнее о настройках реестра см. в [Приложении А](#).

# IndustrialSQL Server и Microsoft SQL Server

IndustrialSQL Server включает в себя систему Microsoft SQL Server. В этой главе описываются отношения между системами IndustrialSQL Server и Microsoft SQL Server с точки зрения сетевого соединения, выборки данных, запуска и управления базой данных.

## Содержание

- [Интеграция с Microsoft SQL Server](#)
- [Сетевое взаимодействие](#)
- [Запуск Microsoft SQL Server](#)
- [Использование SQL Server Enterprise Manager](#)

## Интеграция с Microsoft SQL Server

Microsoft SQL Server является реляционной СУБД (RDBMS). В системах RDBMS информация хранится в отдельных, но соотносящихся таблицах. Сохранение и выборка информации в отдельных таблицах происходит гораздо быстрее, чем при хранении данных в одной большой таблице.

Microsoft SQL Server действует в качестве хранилища данных, обрабатывая запросы клиента с обеспечением точности, целостности и безопасности данных. Эта система также обеспечивает расширенные возможности, такие как автоматическое восстановление, динамическое резервирование и отказоустойчивость, предоставляя из базы данных целостную и содержательную информацию.

Microsoft SQL Server поставляется с интегрированным набором инструментов управления, которые расширяют возможности интерфейса Windows NT. Microsoft SQL Server также интегрируется с программами Performance Monitor, Event Viewer и журналами событий Windows NT для мониторинга всех серверов сети.

IndustrialSQL Server содержит внутри себя Microsoft SQL Server и пользуется всеми возможностями последнего. Подробнее о возможностях Microsoft SQL Server см. в документации Microsoft.

## Сетевое взаимодействие

В этом разделе описывается сетевое взаимодействие между IndustrialSQL Server и Microsoft SQL Server и соединение с клиентами.

Когда установлены IndustrialSQL Server 7.1 и приложения-клиенты, соединение настраивается автоматически; не требуется никаких дополнительных настроек для запуска IndustrialSQL Server или приложений-клиентов при стандартном протоколе named pipes. Однако, если клиент версии IndustrialSQL Server 7.0 связывается с базой данных версии 7.1, следует изменить строку соединения.

[Подробнее см. в главе 12, "Соединение с клиентом".](#)

## Запуск редактора реестра

Для просмотра сетевых связей между IndustrialSQL Server и Microsoft SQL Server можно использовать редактор реестра.

### ➤ Как запустить редактор реестра

1. В меню **Start** выберите **Run**.
2. Появится диалоговое окно **Run**.

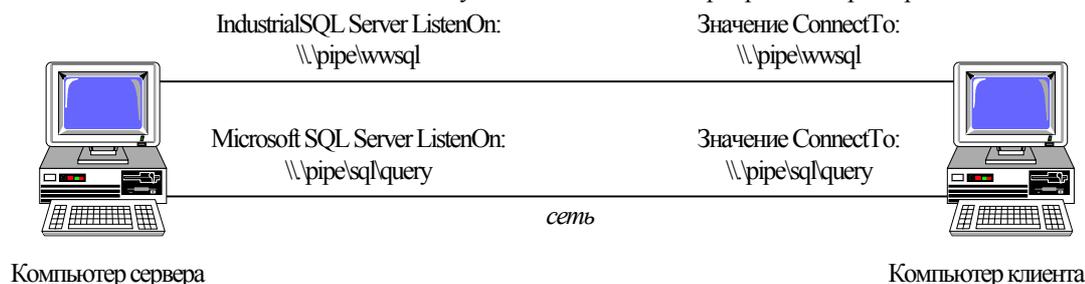


3. Введите **regedt32** и нажмите **OK**.
4. Появится окно **Registry**, показывающее все ключи реестра.



## Соединения с IndustrialSQL Server

После установки IndustrialSQL Server клиенты могут напрямую соединяться с Microsoft SQL Server или с IndustrialSQL Server, который затем устанавливает соединение с Microsoft SQL Server. Стандартное имя канала для соединения с Microsoft SQL Server `\\.\pipe\sql\query`. Стандартное имя канала для соединения с IndustrialSQL Server `\\.\pipe\wwsql`. Настройка IndustrialSQL Server и the Microsoft SQL Server с разными значениями *ListenOn* делает возможной независимую связь с каждым сервером. Например:



Задания Microsoft SQL Server указаны в значении реестра *ListenOn* под ключом:

```
HKEY_LOCAL_MACHINE
  \SOFTWARE
    \Microsoft
      \MSSQLServer
        \MSSQLServer
```

Значение в этом ключе определяет, где Microsoft SQL Server принимает запросы клиентов на соединение. Для связи с Microsoft SQL Server зарегистрируйтесь, используя имя компьютера Microsoft SQL Server в качестве имени сервера.

Задания IndustrialSQL Server указаны в значении реестра *ListenOn* под ключом:

```
HKEY_LOCAL_MACHINE
  \SOFTWARE
    \Microsoft
      \MSSQLServer
        \InSQL
```

Значение в этом ключе определяет, где IndustrialSQL Server принимает запросы клиентов на соединение. После установки IndustrialSQL Server автоматически создается псевдоним "InSQL" и подключается к строке связи `\\.\pipe\wwsql`. Для соединения с IndustrialSQL Server зарегистрируйтесь, используя этот псевдоним в качестве имени сервера.

[Подробнее см. в главе 12, "Соединения с клиентом".](#)

---

**Примечание.** Microsoft SQL Server кэширует полученные ранее данные для последующего использования во время текущего сеанса работы. Если каким-то образом архивные блоки были увеличены (например, при переводе часов на зимнее/летнее время или при копировании архивных блоков), клиент должен рассоединиться и вновь соединиться с сервером для обновления и получения новейших данных.

---

## Настройка IndustrialSQL Server на другие протоколы

Все настройки автоматически производятся для протокола named pipes при установке IndustrialSQL Server. Дополнительные значения *ListenOn* могут быть настроены для IndustrialSQL Server путем добавления требуемых значений в реестр с помощью редактора реестра. Для включения соединения клиента и сервера через другой протокол следует:

1. Настроить значение *ListenOn* в IndustrialSQL Server для каждого типа сети, который будет использоваться (кроме настроек сетевых библиотек Multi-Protocol).
2. Настроить значение *ListenOn* в Microsoft SQL Server для нового протокола.
3. Настроить значение *ConnectTo* на каждом компьютере клиента, кроме тех случаев, когда применимы стандартные идентификаторы процесса. После установки инструментов клиента IndustrialSQL Server значения *ConnectTo* для компьютера клиента автоматически настраиваются на протокол named pipes.

Например, если требуется соединение клиента через TCP/IP, добавьте сетевую библиотеку Windows Sockets, идентификатор узла и идентификатор процесса в значение ключа *ListenOn* реестра IndustrialSQL Server. Для IPX добавьте сетевую библиотеку NetWare и имя IPX Transport в значение ключа *ListenOn* реестра IndustrialSQL Server.

При загрузке IndustrialSQL Server проверяет реестр для определения, какие сетевые библиотеки следует загружать и какие параметры пересылать клиенту, подающему запрос. Если сервер не определен в реестре, то протокол по умолчанию устанавливается на сетевую библиотеку named pipes, а идентификатор процесса по умолчанию устанавливается на стандартный канал сервера SQL `\\.\pipe\sql\query`.

## Соединение через Multi-Protocol

Для соединений клиентов, настроенных на использование сетевой библиотеки Multi-Protocol, никакое имя службы сервера *ListenOn* не обязано быть указано при загрузке SQL-сервером SSMSRPCN.DLL. По умолчанию клиенты устанавливают соединение просто по имени компьютера. При установке соединения сетевая библиотека посылает имя компьютера в RPC, определяющее, какие протоколы IPC доступны, и пытается использовать один из них, пока связь не будет установлена. (Считаются проверенными и поддерживаются только NWLink IPX/SPX, гнезда TCP/IP и named pipes).

## Соединения с клиентами

IndustrialSQL Server является высококачественной системой управления реляционной базой данных, облегчающей доступ к огромным объемам производственных данных в среде взаимодействия между сервером и клиентами. Интерфейс сервер/клиент поддерживает библиотеку БД и предоставляет API-интерфейс клиента ODBC для работы с базой данных сервера SQL.

IndustrialSQL Server поддерживает много типов приложений-клиентов, работающих в различных операционных системах, и поддерживает одновременные соединения с клиентами через:

- Named Pipes
- IPX/SPX
- Гнезда TCP/IP
- Multi-Protocol

Подробнее о настройке клиентов для работы с этими протоколами см. в главе 12, "Соединения с клиентами".

## Запуск Microsoft SQL Server

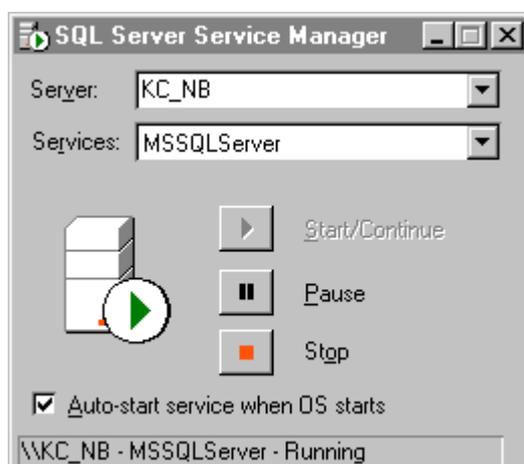
Один из способов запуска встроенного Microsoft SQL Server заключается в использовании программы SQL Server Service Manager. Эта программа устанавливается вместе с Microsoft SQL Server. Программа InSQL Control, используемая для запуска системы IndustrialSQL Server, запустит также и Microsoft SQL Server.

Подробнее о запуске системы IndustrialSQL Server см. в главе 5, "InSQL Control".

### ➤ Как запустить Microsoft SQL Server:

1. В меню **Start** выберите группу программ Microsoft SQL Server и нажмите SQL Service Manager.

Появится окно **SQL Service Manager**.



2. В списке **Server** выберите стандартное имя сервера Microsoft SQL Server.
3. Нажмите **Start/Continue** для запуска службы. Состояние службы отображается внизу окна.

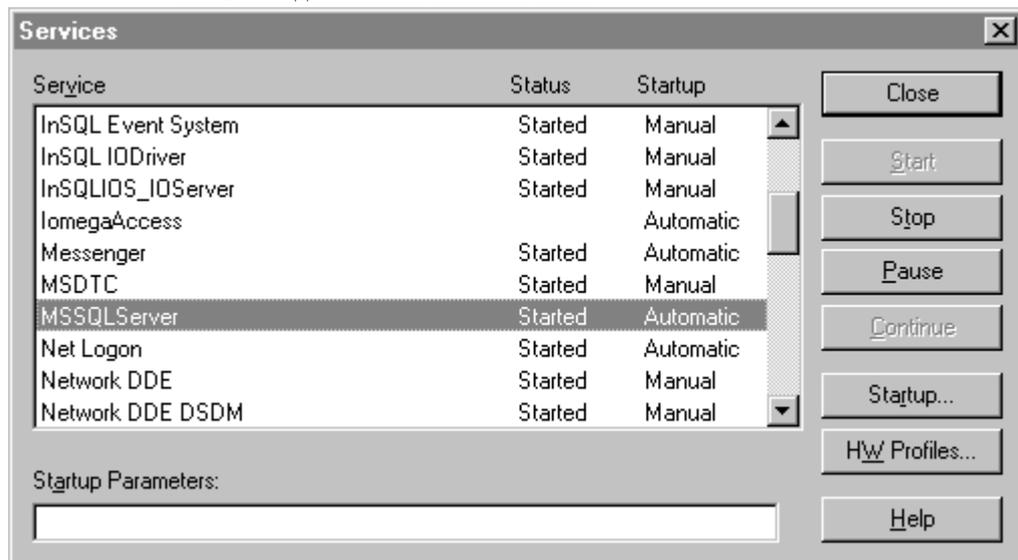
## Настройка автоматической загрузки

Microsoft SQL Server может автоматически запускаться при загрузке компьютера. Этот параметр можно выбрать во время установки Microsoft SQL Server или настроить через Панель управления.

➤ **Как настроить автозагрузку:**

1. В меню **Start** выберите **Settings**, затем нажмите **Control Panel**.
2. Дважды нажмите левой кнопкой мыши на ярлык **Services**.

Появится диалоговое окно **Services**.



3. Нажмите кнопку **Startup**.

Появится диалоговое окно **Service**.



4. Выберите тип запуска **Automatic** и нажмите **ОК**.

## Использование SQL Server Enterprise Manager

Как администратор, вы, скорее всего, большую часть своего времени потратите на работу с Microsoft SQL Server через программу SQL Server Enterprise Manager. Программа SQL Server Enterprise Manager позволяет управлять теми компонентами Microsoft SQL Server, которые используются в IndustrialSQL Server.

С помощью SQL Server Enterprise Manager можно выполнять следующие функции:

- Регистрировать серверы
- Управлять резервированием
- Управлять базами данных
- Управлять устройствами
- Управлять учетными записями и допуском
- Управлять таблицами, представлениями, хранимыми процедурами, триггерами, индексами, правилами, стандартами и пользовательскими типами данных
- Планировать задания
- "Перетаскивать" объекты с одного сервера на другой или внутри сервера
- Создавать сценарии SQL

Для запуска SQL Server Enterprise Manager выберите в меню **Start** Windows группу программ Microsoft SQL Server и нажмите SQL Server Enterprise Manager.

## Регистрация сервера

При первом запуске SQL Server Enterprise Manager никаких серверов в окне серверов не появится. Для управления сервером через SQL Server Enterprise Manager следует его зарегистрировать. При регистрации сервера вы сообщаете программе SQL Server Enterprise Manager логическое имя и учетную запись пользователя для связи с базой данных Microsoft SQL Server.

### ➤ Как зарегистрировать сервер:

1. Запустите SQL Server Enterprise Manager.
2. В меню **Action** выберите **New SQL Server Registration**.
3. Появится мастер **Register SQL Server Wizard**.
4. Следуйте инструкциям мастера для прохождения регистрации.

См. подробности в документации Microsoft.

## Ориентация в программе SQL Server Enterprise Manager

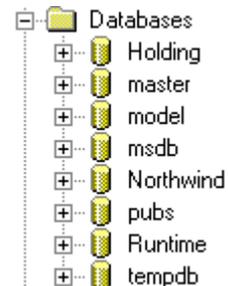
Чтобы управлять сервером через SQL Server Enterprise Manager, необходимо установить соединение с ним. Около значков серверов, с которыми установлено соединение, появится красная зигзагообразная линия.



После того, как связь с сервером установлена, вы сможете просмотреть полное дерево консоли сервера, включая все устройства и базы данных. Можно открывать папки так же, как в Проводнике Windows.

### Управление файлами базы данных

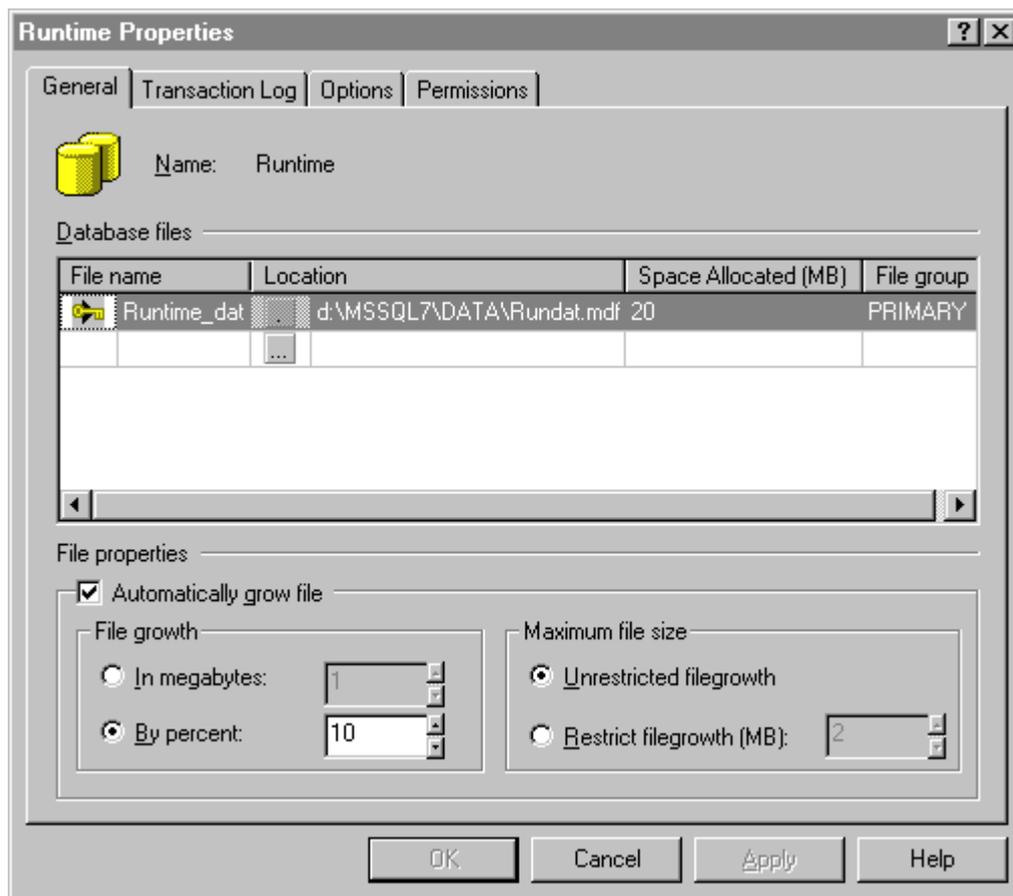
При открытии папки **Database** вы можете просмотреть базы данных, созданные в Microsoft SQL Server, а также базы данных, созданные в IndustrialSQL во время установки.



➤ **Как работать с файлами баз данных:**

1. Нажмите правой кнопкой мыши на базе данных в дереве консоли.
2. При появлении меню нажмите **Properties**.

Появится диалоговое окно **Properties**.



3. Диалоговое окно **Properties** содержит информацию, такую как путь и размер файлов баз данных и соответствующий журнал транзакций. Здесь можно устанавливать параметры базы данных и отменять допуск для пользователей базы данных.

Подробнее об управлении файлами базы данных см. в документации Microsoft.

4. Нажмите **Cancel**, чтобы закрыть диалоговое окно.

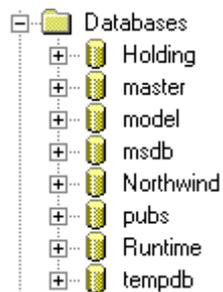
## Управление базами данных

Когда вы открываете папку **Database**, вы можете просмотреть все базы данных сервера.

---

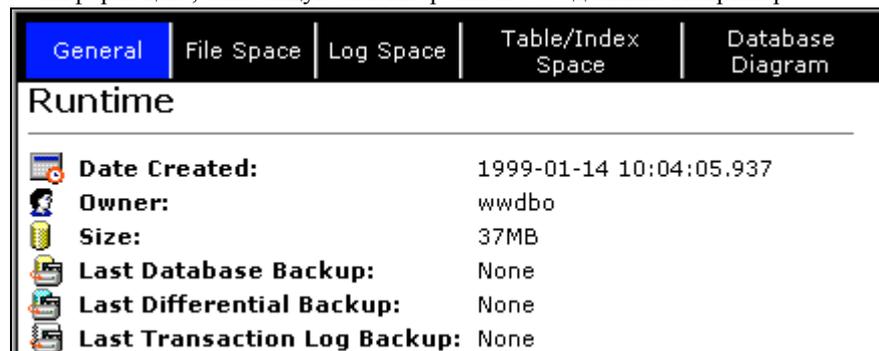
**Примечание.** Не следует редактировать какие-либо готовые таблицы, входящие в комплект IndustrialSQL Server.

---



1. Выберите базу данных в дереве консоли.

Панель задач (область справа от окна сервера) отобразит основную информацию, относящуюся к выбранной базе данных. Например:



2. Для выполнения конкретного задания в базе данных нажмите правой кнопкой мыши на значок базы данных в дереве консоли.
3. В появившемся меню выберите **Tasks** и затем выберите команду меню, соответствующую нужному заданию.

---

**Примечание.** Microsoft SQL Server включает в себя несколько мастеров, которые помогают выполнять самые простейшие задания. Для использования этих мастеров выберите **Tools** в контекстном меню и нажмите **Wizards**.

---

Подробнее см. документацию Microsoft.

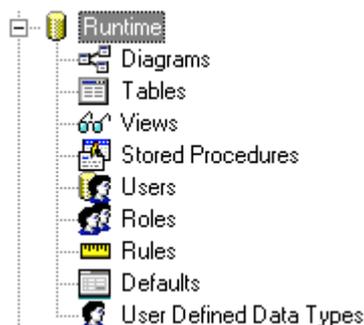
## Управление объектами базы данных

Объект базы данных — это компонент базы данных: таблица, индекс, триггер, представление, ключ, запрет, стандарт, правило, пользовательский тип данных или хранимая процедура. Все, что имеет имя и занимает место в базе данных, называется объектом.

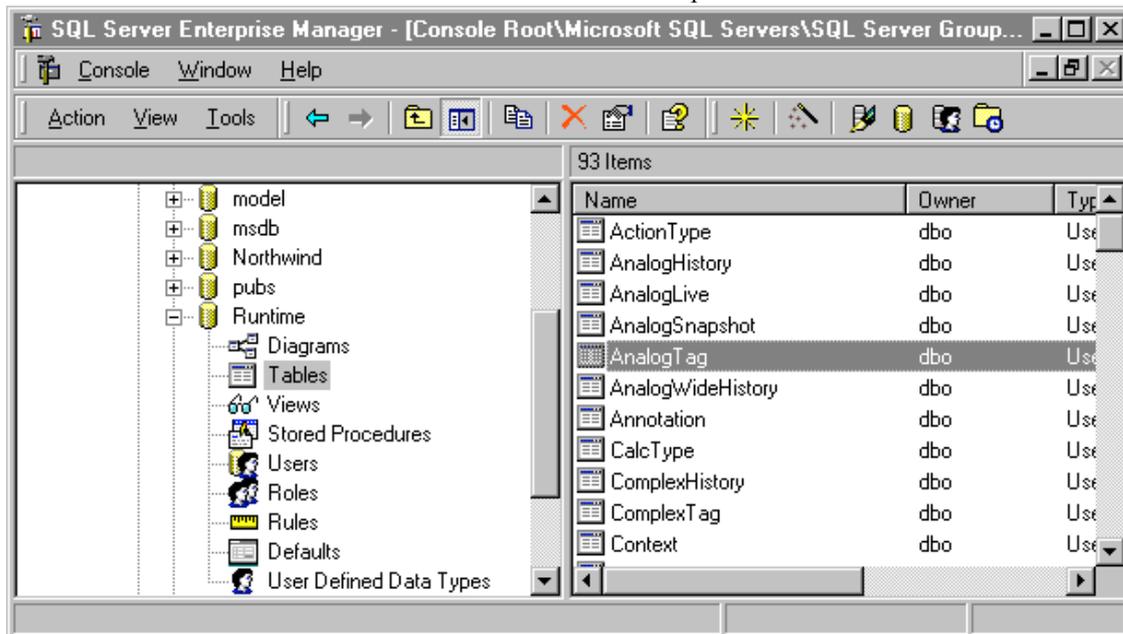
**Примечание.** Не следует редактировать какие-либо готовые таблицы, входящие в комплект IndustrialSQL Server.

### ➤ Как работать с объектом базы данных:

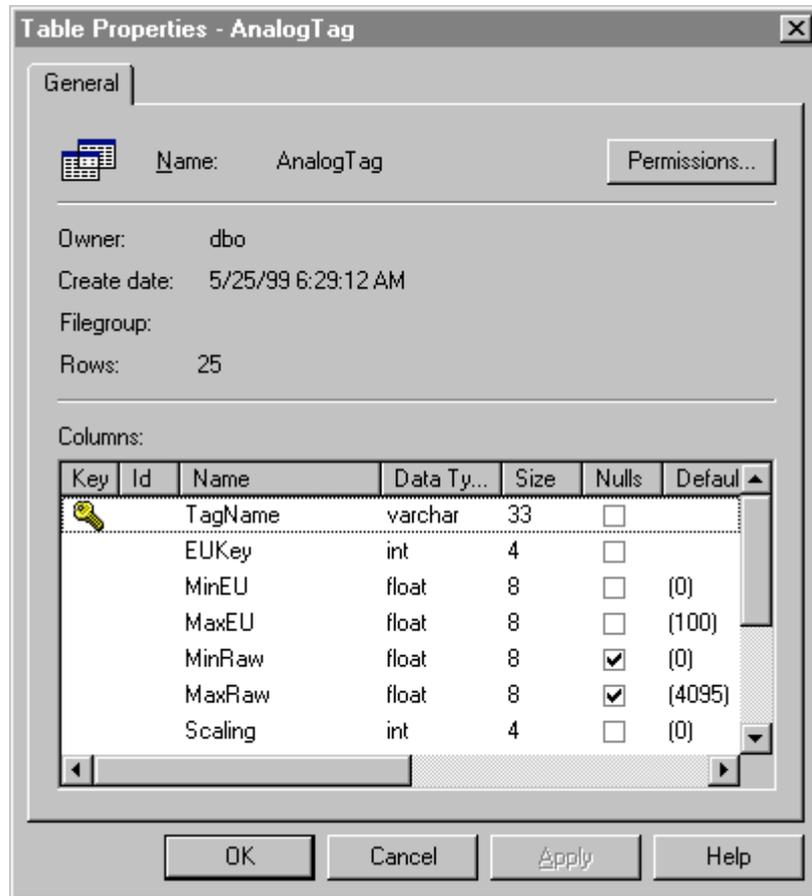
1. Откройте папку базы данных **Runtime**.



2. Все группы объектов базы данных появятся под папкой **Runtime**. Для просмотра всех объектов в группе просто выберите группу в дереве консоли.
3. Для управления любым объектом базы данных просто дважды нажмите левой кнопкой мыши на объект в правой панели окна:



4. Появится диалоговое окно управления объектом. Например, диалоговое окно **Table Properties** появится, когда вы дважды нажмете левой кнопкой мыши на таблице IndustrialSQL Server.



5. Выберите **Cancel**.

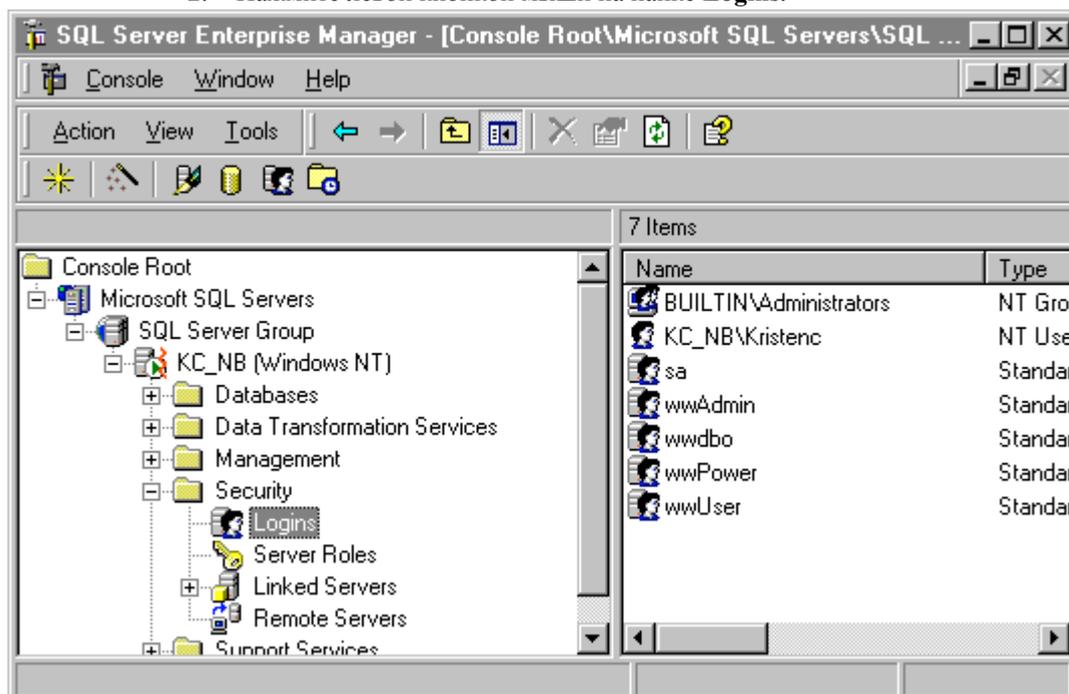
Подробнее см. документацию Microsoft.

## Управление учетными записями

В дополнение к учетным записям Microsoft SQL Server, в IndustrialSQL Server созданы четыре дополнительные стандартные учетные записи: **wwAdmin**, **wwdbo**, **wwPower** и **wwUser**.

➤ **Как работать с учетными записями:**

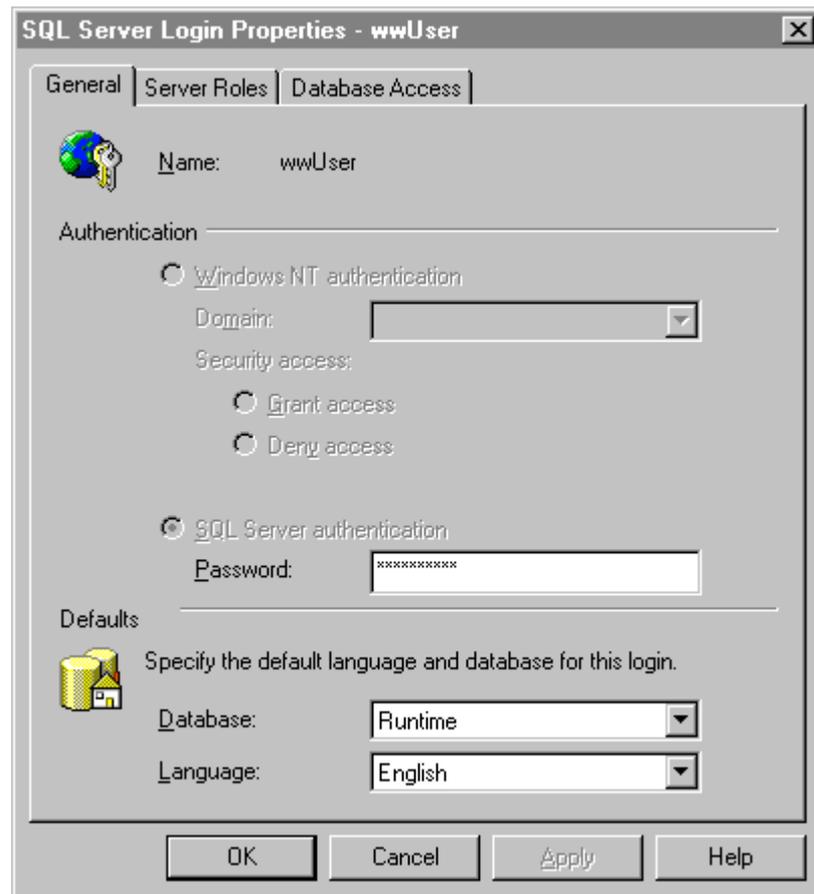
1. Откройте папку **Security** под сервером в дереве консоли.
2. Нажмите левой кнопкой мыши на папке **Logins**.



Стандартные учетные записи появятся в правой панели окна.

3. Дважды нажмите левой кнопкой мыши на учетной записи, свойства которой хотите просмотреть.

Появится диалоговое окно **SQL Server Login Properties**.



4. Если вы системный администратор (sa), то вы можете добавлять, изменять и удалять учетные записи. Подробнее см. документацию Microsoft.

## Соединения с клиентами

В этой главе описывается утилита SQL Server Client Network и объясняется, как обеспечить соединение клиентов с IndustrialSQL Server через механизм межпроцессовой связи (IPC) "named pipes", сетевые протоколы Windows IPX/SPX и TCP/IP или multi-protocol. Также описываются шаги, необходимые для настройки соединений клиентов с использованием любой сетевой библиотеки, поддерживаемой Microsoft. А именно:

- Named Pipes
- IPX/SPX
- TCP/IP Sockets
- Multi-Protocol

Эта глава подразумевает, что вы уже знакомы с основными сетевыми механизмами и протоколами, такими как TCP/IP и named pipes, а также с концепцией сетевых адресов. Также вы должны понимать роль библиотек БД и сетевых библиотек в установке соединений клиента/сервера и выборке данных.

Подробнее об установке клиентов IndustrialSQL Server см. в Руководстве администратора системы FactorySuite.

Подробнее о настройке соединений клиентов см. в документации Microsoft.

### Содержание

- Соединения клиентов
- Утилита SQL Server Client Network
- Настройка общих параметров клиента
- Настройка соединений клиента
- Устранение проблем соединения клиентов с IndustrialSQL
- Правила настройки клиента ODBC

## Соединения клиентов

В зависимости от типа сети, используемого для подключения к IndustrialSQL Server, может возникнуть необходимость настройки компьютеров-клиентов. Настройка клиента включает в себя установки переменных среды, параметров или значений соединения. Эти настройки клиента хранятся в реестре.

Когда инструменты клиента настраиваются для соединения с системой IndustrialSQL Server по протоколу named pipes (по умолчанию), не требуется никакой дополнительной настройки. Однако, если устанавливаются другие инструменты клиента для работы с IndustrialSQL Server, или если вы изменяете стандартный сетевой протокол для клиентов, следует произвести настройку клиента или сетевого протокола. В этой главе описывается утилита SQL Server Client Network и способы настройки соединений клиента для каждого сетевого протокола.

Настройка соединений клиента подразумевает:

1. Указание сервера, с которым клиент будет соединяться.
2. Выбор сетевого протокола для связи с сервером.
3. Указание идентификата сети, называемого строкой соединения с указанным сервером.

## Строка сетевого соединения

Каждый программный процесс в сети должен иметь уникальную идентификацию. Набор идентификаторов, которые уникально идентифицируют программные процессы в сети, задается для каждого сетевого протокола. Этот набор идентификаторов, известный как строка соединения, может отличаться в зависимости от протокола. Например, строка соединения для протокола named pipes отличается от строки соединения по протоколу TCP/IP.

## Настройки реестра Windows NT

Утилита SQL Server Client Network добавляет информацию настройки под следующим ключом реестра:

```
HKEY_LOCAL_MACHINE
  \SOFTWARE
    \Microsoft
      \MSSQLServer
        \Client
          \ConnectTo
```

Для каждого соединения сервера/клиента запись в реестре включает в себя имя значения, тип данных и строку данных. Имя значения определяет имя сервера, тип данных — это REG\_SZ, а строка данных включает в себя имя сетевой библиотеки и строку соединения.

## Утилита SQL Server Client Network

Для настройки связи клиента с сервером используется утилита SQL Server Client Network, устанавливаемая вместе с Microsoft SQL Server. С помощью этой утилиты можно:

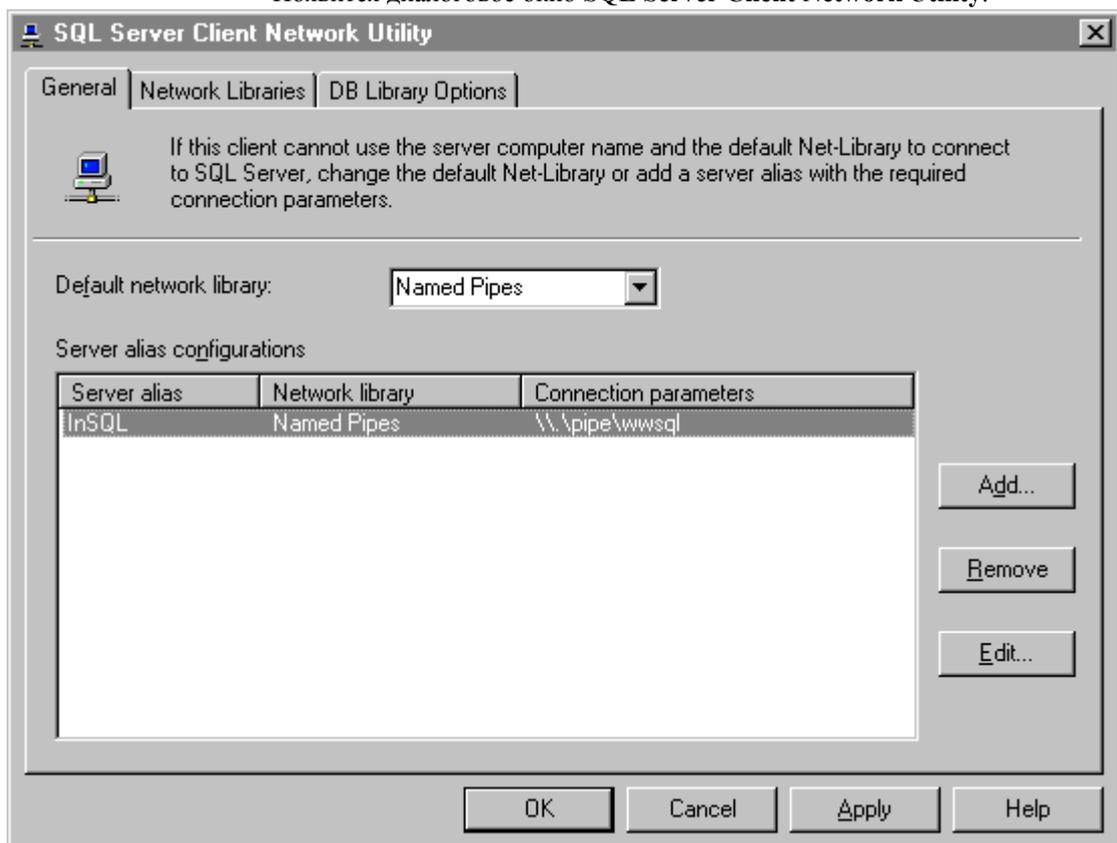
- Узнать информацию о версии библиотеки БД.
- Указать, где клиент устанавливает соединение с сервером SQL.

**Примечание.** Для клиентов, установленных посредством программы установки IndustrialSQL Server и использующих стандартный протокол named pipes, не требуется дополнительной настройки.

### ➤ Как запустить утилиту SQL Server Client Network:

1. В меню Windows **Start** выберите папку **Programs** и затем выберите папку Microsoft SQL Server.
2. Выберите **Client Network Utility** из списка доступных инструментов сервера SQL.

Появится диалоговое окно **SQL Server Client Network Utility**.



Панель **General** позволяет указать соединение с IndustrialSQL Server.

---

**Примечание.** Для соединения с Microsoft SQL Server псевдонимы не настраиваются. Это происходит из-за того, что соединение автоматически управляется в сети, если в качестве имени сервера при регистрации используется имя компьютера Microsoft SQL Server.

---

После установки IndustrialSQL Server создается псевдоним "InSQL" для соединения с IndustrialSQL Server, который использует следующую строку соединения:

```
\\.\pipe\wwsql
```

Если клиент соединяется с Industrial SQL Server, который установлен на том же компьютере, имя сервера указывается периодом (\\.) в строке соединения. Если вы хотите указать другой компьютер, вставьте в строку имя компьютера, на котором установлен сервер. Например:

```
\\InSQL1\pipe\wwsql
```

Каждый псевдоним содержит запись под следующим ключом реестра:

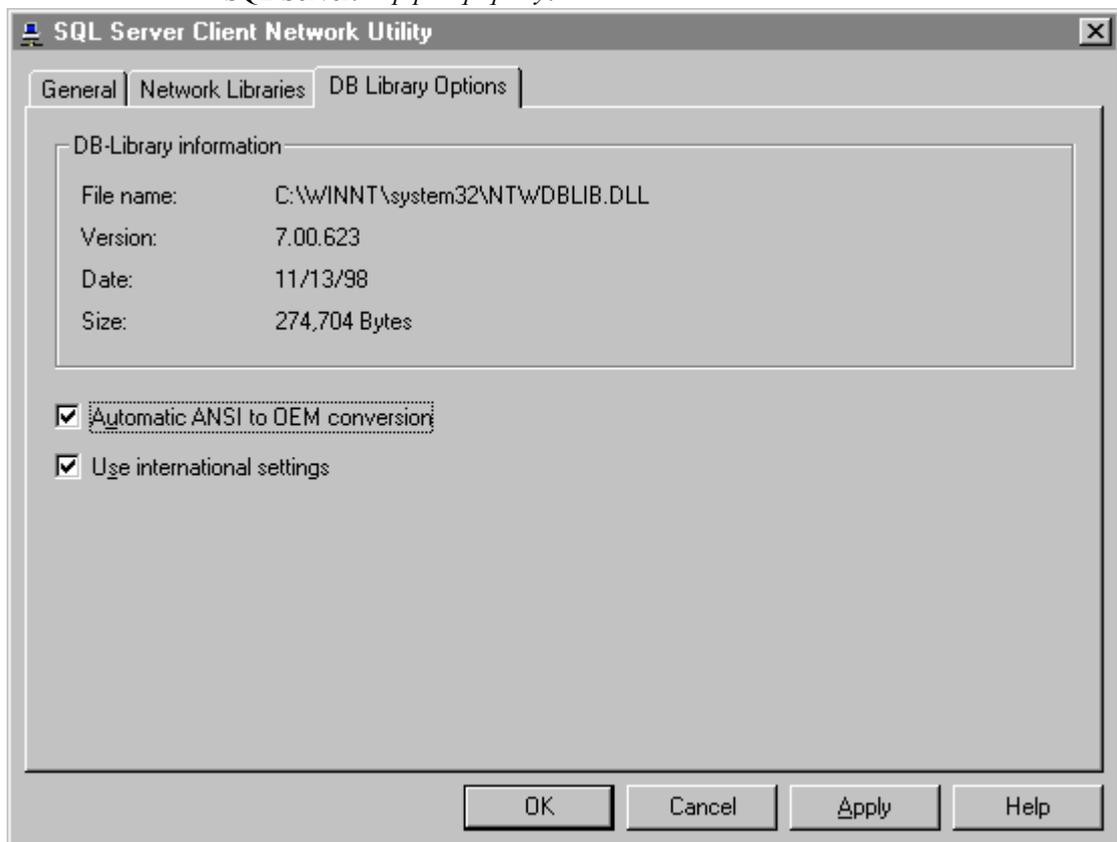
```
HKEY_LOCAL_MACHINE
  \SOFTWARE
    \Microsoft
      \MSSQLServer
        \Client
          \ConnectTo
```

## Параметры библиотеки БД

Библиотекой БД называется набор языковых библиотек, создающих интерфейс программирования приложения (API) для сетевых клиентов. Панель **DB Library Options** показывает версию библиотеки, используемой клиентами для соединения с SQL Server. К этим клиентам относятся утилиты SQL Server и приложения, устанавливающие соединение через ODBC.

Когда клиент библиотеки БД соединяется с сервером, он немедленно проверяет переменные среды, параметры или значения соединения на наличие определения сетевого соединения с сервером.

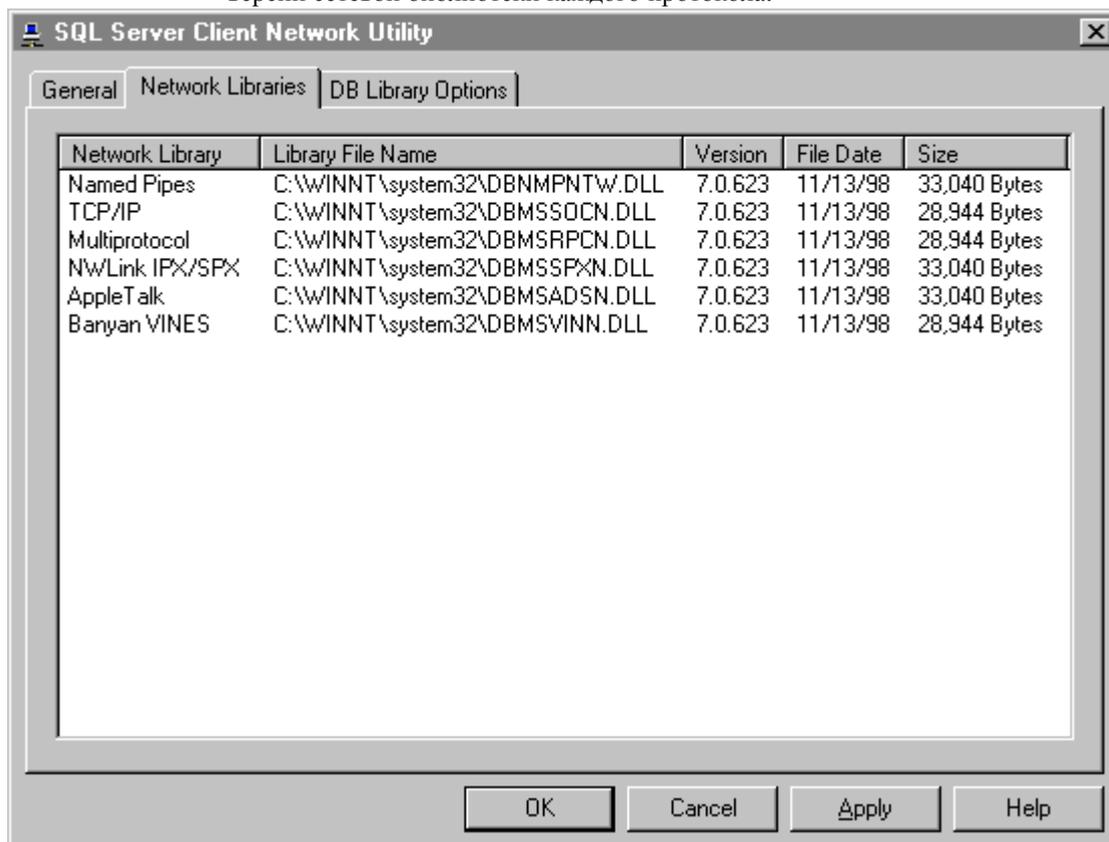
Если сетевое соединение с сервером не определено, клиент библиотеки БД пытается соединиться с сервером, используя строку соединения для Microsoft SQL Server: `\\.\pipe\sql\query`.



## Сетевая библиотека

Сетевой библиотекой называется набор функций, управляющий сетевым соединением и маршрутизацией. Каждый сетевой протокол имеет собственную сетевую библиотеку, которая используется компьютерами для общения по сети.

На панели **Network Libraries** можно просмотреть информацию о текущей версии сетевой библиотеки каждого протокола.



## Создание нового псевдонима

Можно создать псевдоним для IndustrialSQL Server. В приведенном примере псевдоним "InSQLServer1" добавлен для сервера, запущенного на сетевом компьютере "слушающем" с адреса `\\InSQL01\pipe\wwsql`. Если вы создаете псевдоним для соединения, обязательно укажите имя компьютера IndustrialSQL Server вместо стандартной точки (.).

---

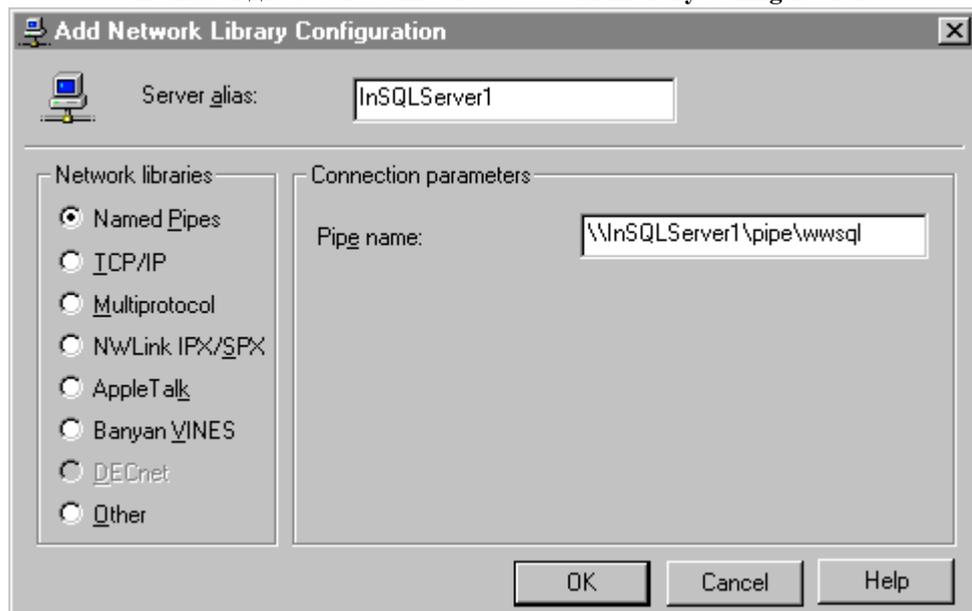
**Примечание.** Для того, чтобы приложения-клиенты IndustrialSQL Server 7.0 могли соединяться с базой данных IndustrialSQL Server 7.1, следует заменить "sql" на "wwsql" в строке соединения для всех используемых клиентами псевдонимов.

---

➤ **Как создать новый псевдоним:**

1. В утилите SQL Server Client Network выберите панель **General**.
2. В списке **Default network library** выберите **Named Pipes**.
3. Нажмите **Add**.

Появится диалоговое окно **Add Network Library Configuration**.



4. В поле **Server alias** введите псевдоним, который вы хотите использовать.  
Псевдонимом может быть имя компьютера, на котором установлен IndustrialSQL Server, или другое имеющее смысл имя.
5. В поле **Pipe name** измените часть строки "sql" на "wwsql."
6. Если IndustrialSQL Server установлен на удаленном сетевом компьютере, замените в строке соединения точку (.) на имя компьютера.
7. Нажмите **OK**.

## Настройка общих параметров клиента

Прежде чем настраивать соединения клиента для указанных протоколов, следует сначала настроить некоторые общие параметры клиента/сервера. Эти параметры описаны ниже. Подробнее о любом из этих параметров см. в документации Microsoft.

1. Определите версии библиотек и драйвер ODBC-сервера SQL.

Важно, чтобы клиент запускал текущие версии и библиотек БД, и сетевых библиотек. Проверьте, правильны ли номера версий библиотек. Также следует проверить, использует ли Microsoft SQL Server правильный драйвер ODBC.

2. Настройте стандартную сетевую библиотеку.

После установки Microsoft SQL Server стандартной сетевой библиотекой является named pipes. Если вы хотите изменить стандартную сетевую библиотеку для поддержки другого механизма межпроцессовой связи (IPC), следует изменить этот параметр. Воспользуйтесь инструкциями, приводимыми в документации Microsoft.

3. Настройте параметры библиотеки БД.

Библиотеку БД можно настроить на автоматическое изменение символов OEM на ANSI при общении клиента с сервером SQL, и символы ANSI на символы OEM при общении сервера SQL с клиентом. Также можно настроить, будет ли библиотека БД считывать форматы даты, времени и валюты из системы.

## Настройка соединений клиента

IndustrialSQL Server поддерживает сетевые соединения, устанавливаемые на базе средств Windows 95, Windows 98 и Windows NT. IndustrialSQL Server может соединяться с любым 32-битным клиентом через любой сетевой протокол, для которого установлена сетевая библиотека.

---

**Примечание.** Поддерживаются только соединения клиентов на основе Windows 95, Windows 98 и Windows NT (32-битные).

---

Вы можете настроить соединения сетевого клиента, используя следующие сетевые библиотеки:

- Named Pipes
- TCP/IP Sockets
- NetWare IPX/SPX
- Multi-Protocol

## Протокол Named Pipes

Протокол named pipes использует строку соединения из двух частей для уникальной идентификации процессов сервера. В строке соединения для протокола named pipes требуется указать идентификатор узла и идентификатор процесса. Идентификатором узла является имя компьютера NetBIOS для IndustrialSQL Server. Примером идентификатора узла для протокола named pipes служит:

```
\\InSQLServer8
```

Идентификатором процесса является имя канала. Примером канала является

```
\pipe\wwsql
```

Все элементы среды Microsoft SQL Server используют стандартный идентификатор процесса, основанный на стандартном имени канала Microsoft SQL Server, которое выглядит так:

```
\\имякомпьютера\pipe\sql\query
```

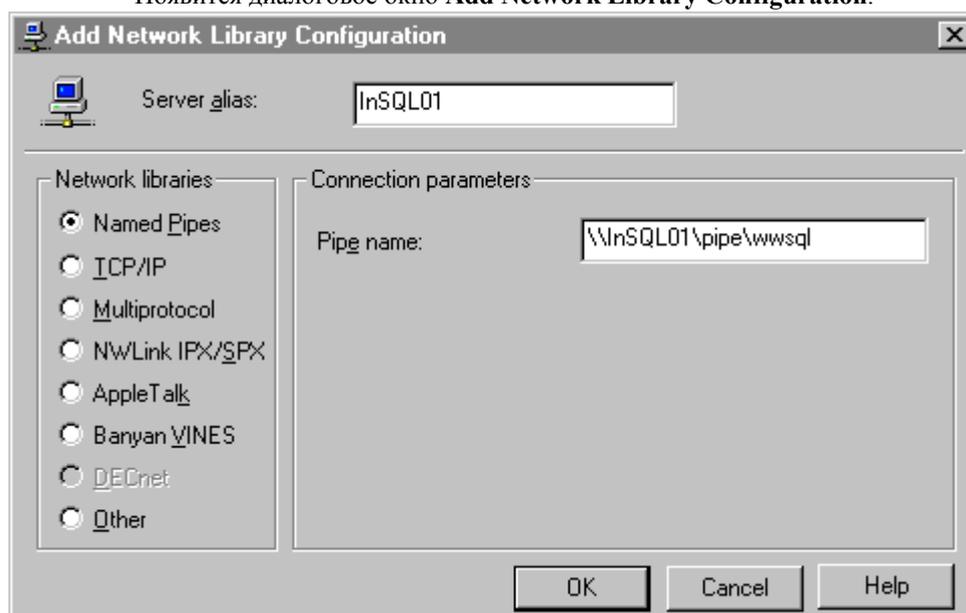
Стандарт позволяет клиентам библиотеки БД обращаться к нескольким серверам, не нуждаясь в отдельной настройке каждого компьютера.

## Настройка клиентов на протокол Named Pipes

### ➤ Как настроить клиента на протокол named pipes:

1. Запустите утилиту SQL Server Client Network.
2. Выберите панель **General**.
3. Нажмите **Add**.

Появится диалоговое окно **Add Network Library Configuration**.



4. Введите следующую информацию для настройки клиента:

**Server alias:** Уникальное имя компьютера IndustrialSQL Server. Используется приложениями-клиентами для опознавания IndustrialSQL Server в сети.

**Network Library:** Named Pipes

**Connection String:** `\\имякомпьютера\pipe\wssql`

Где *имякомпьютера* — это имя компьютера, на котором запущен IndustrialSQL Server.

Заметьте, что выражение "sql" в строке должно быть заменено на "wssql".

5. Нажмите **ОК** для добавления новой настройки клиента в список текущих записей.

## Протокол NetWare IPX/SPX

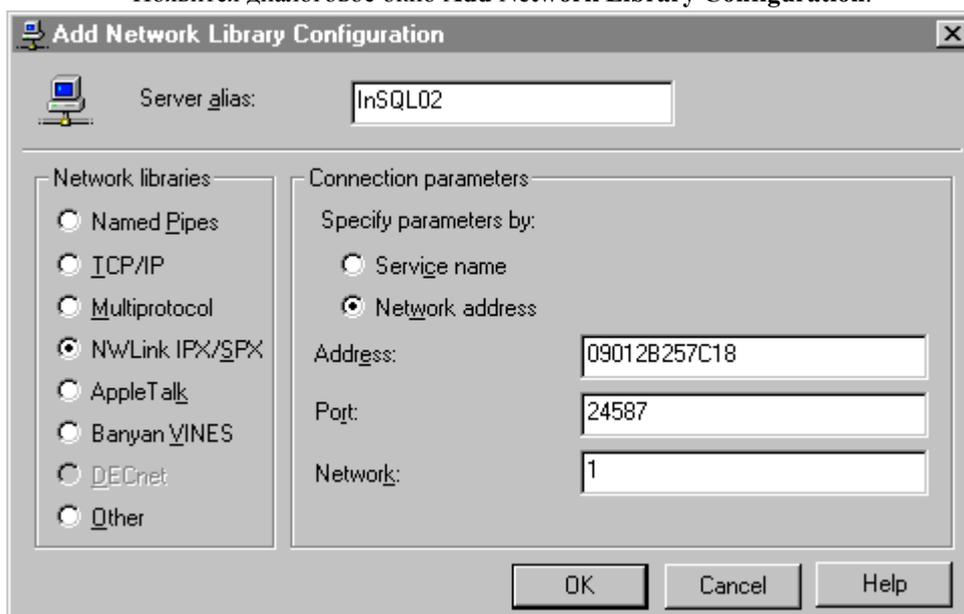
Протокол NetWare IPX/SPX использует идентификатор с одной частью, который является идентификатором узла. Идентификатор узла адреса транспортировки IPX обычно указывает на имя компьютера. Запросы NetWare (перенаправления) представляют собой широковещательные пакеты, рассылаемые по сети один раз в минуту. Когда адрес узла (адрес транспортировки IPX) предоставлен, клиенты NetWare могут установить соединение с серверами, передавая адрес транспортировки IPX компьютеру сервера.

## Настройка клиентов NetWare IPX/SPX

➤ **Как настроить соединение клиента по IPX/SPX:**

1. Запустите утилиту SQL Server Client Network.
2. Выберите панель **General**.
3. Нажмите **Add**.

Появится диалоговое окно **Add Network Library Configuration**.



4. Введите следующую информацию для настройки клиента:

**Server alias:** Уникальное имя сервера. Это имя используется приложениями-клиентами для опознавания сервера в сети.

**Network Library:** NWLink IPX/SPX

**Connection String:** *Имя службы*

где *имя службы* — это имя компьютера Microsoft Windows NT®, на котором запущен сервер SQL.

*Адрес:Порт:Сеть*

где *Адрес:Порт:Сеть* — это адрес транспортировки IPX, настроенный для работы с IndustrialSQL Server. Обычно (необязательно) это компьютер, на котором запущен IndustrialSQL Server.

5. Нажмите **ОК** для добавления новой настройки клиента в список текущих записей.

## Протокол Multi-Protocol

Соединения через Multi-Protocol позволяют одновременно использовать несколько механизмов IPC и используют технологию Windows NT RPC (вызов удаленной процедуры). Если для получения имен через поддерживаемые протоколы используется локальная база данных RPC, то в строке соединения необязательно указывать порты и гнезда.

---

**Примечание.** Проверенными и надежными считаются только протоколы TCP/IP Windows Sockets, NWLink IPX/SPX и Named Pipes.

---

В случае multi-protocol, для разрешения имен клиентом используется имя компьютера сервера .

## Настройка клиента на Multi-Protocol

Удостоверьтесь, что ваш клиент поддерживает хотя бы один из протоколов IPC, поддерживающих Multi-Protocol на сервере (Named Pipes, IPX/SPX, TCP/IP или Windows Sockets). С помощью параметров Advanced утилиты SQL Server Client Network клиенты на базе Windows и Windows NT могут создавать псевдонимы имен серверов для использования сетевой библиотеки Multi-Protocol.

---

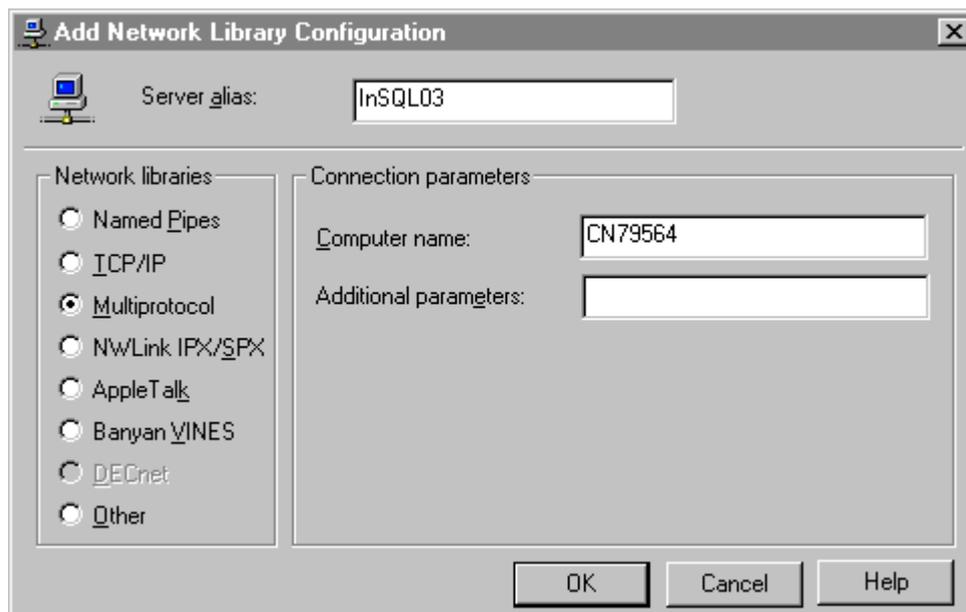
**Примечание.** Пользователям Windows NT: если вы используете SPX/IPX через NWLink, следует установить также службу Windows NT NetWare Client Services. Подробнее о службе Windows NT NetWare Client Services см. в документации Microsoft.

---

### ➤ Как настроить соединение клиента по Multi-Protocol:

1. Запустите утилиту SQL Server Client Network.
2. Выберите панель **General**.
3. Нажмите **Add**.

Появится диалоговое окно **Add Network Library Configuration**.



4. Введите следующую информацию для настройки клиента:

**Server alias:** Обычно, это имя компьютера, на котором запущен IndustrialSQL Server и идет принятие запросов сетевой библиотеки multi-protocol. Однако, если используется иное имя (не имя компьютера), следует ввести определенную строку соединения.

Подробнее о разрешении имен см. в документации Microsoft.

**Network Library:** Выберите **Multiprotocol**.

**Connection String:** Следует оставить это поле пустым, если в качестве имени компьютера используется имя сервера. Если же имя сервера не то же самое, что имя компьютера, имя компьютера должно быть указано в строке соединения.

## Протокол TCP/IP

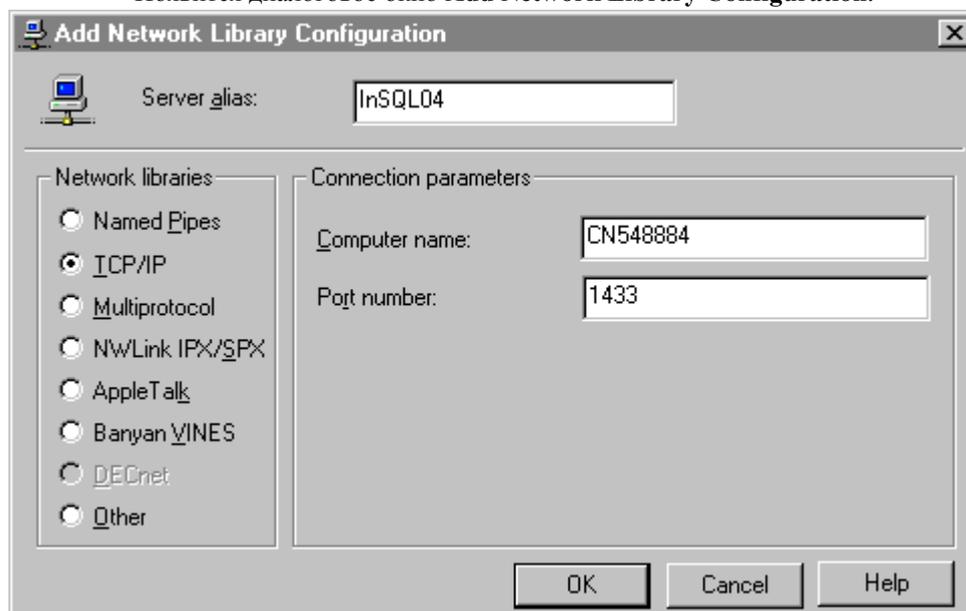
Протокол TCP/IP использует идентификатор из двух частей для уникальной идентификации процесса сервера: идентификатор узла и идентификатор процесса. Для сетей, использующих адреса гнезд TCP/IP, идентификатором узла является цифровой IP-адрес. Например, 11.1.5.123. Идентификатор процесса — это номер гнезда; например: 3180.

## Настройка клиентов на TCP/IP

### ➤ Как настроить клиента на TCP/IP:

1. Запустите утилиту SQL Server Client Network.
2. Выберите панель **General**.
3. Нажмите **Add**.

Появится диалоговое окно **Add Network Library Configuration**.



4. Введите следующую информацию для настройки клиента:

**Server alias:** Имя соединения, указываемое клиентом для доступа к IndustrialSQL Server.

**Network Library:** TCP/IP

**Connection String:** *имякомпьютера,[номерпорта]*

где *имякомпьютера* — это сетевой IP-адрес компьютера, на котором запущен IndustrialSQL Server, а *номерпорта* (необязательно) — это номер приложения IndustrialSQL Server на компьютере сервера. По умолчанию сервер SQL получает номер — 1433. Он может быть изменен.

5. Нажмите **OK** для добавления новой настройки клиента в список текущих записей.

## Устранение проблем соединения клиентов с IndustrialSQL

В этом разделе описаны соединения клиентов с базами данных IndustrialSQL Server.

### Клиенты IndustrialSQL Server 7.0.

Для соединения с IndustrialSQL Server 7.1 через клиентов версии 7.0 (InSQL Trend, InSQL QuickLook, InSQL VectorMaster и VectorViewer и Industrial Workbook) требуется исправить строку соединения на компьютере клиента. Это изменение необходимо, потому что IndustrialSQL Server 7.1 более не ожидает запросов по стандартному каналу сервера, на котором он находится; Microsoft SQL Server 7.0 ожидает запросов по стандартному каналу.

---

**Примечание.** По причине того, что схема базы данных изменилась, следует обновить InSQL Configure до версии 7.1 для соединения с IndustrialSQL Server 7.1.

---

➤ **Как изменить строку соединения:**

1. На компьютере клиента запустите утилиту SQL Server Client Network. (В системе Microsoft SQL Server 6.5 эта программа называлась Client Configuration Utility).
2. Измените строку соединения для любых псевдонимов IndustrialSQL Server (стандартом для версии 7.0 было "InSQL") на `\\<имякомпьютера>\pipe\wwsql`, где `<имякомпьютера>` — это имя компьютера, на котором установлен IndustrialSQL Server.
3. Если вы хотите использовать имя компьютера при связи клиента IndustrialSQL 7.0 с IndustrialSQL Server, можно создать псевдоним с тем же именем, что и имя компьютера. Исправьте строку соединения на `\\<имякомпьютера>\pipe\wwsql`.

---

**Примечание.** Может не возникнуть надобности создавать псевдоним с тем же именем, что и у Microsoft SQL Server, если несколько клиентов на том же компьютере используют имя компьютера для прямого соединения с Microsoft SQL Server.

---

4. Для прямого соединения клиента с Microsoft SQL Server создайте псевдоним и установите строку соединения на `\\<имякомпьютера>\pipe\sql\query`.

## ODS-клиенты IndustrialSQL Server 7.1

К клиентам открытых служб данных (ODS) IndustrialSQL Server 7.1 относятся: InSQL Trend, ActiveTrend, InSQL VectorMaster, VectorViewer и InSQL QuickLook. Эти клиенты могут соединяться только с IndustrialSQL Server версии 7.1 или более поздней. Эти клиенты соединяются, используя строку соединения IndustrialSQL Server, а не строку соединения Microsoft SQL Server. Эти клиенты "сходу" создают подходящую строку соединения, основываясь на имени сервера, указанном в диалоговом окне регистрации. [Подробнее о сервере базы данных ODS IndustrialSQL Server см. в главе 6, "Выборка данных"](#).

Шаги, предпринимаемые клиентом для создания строки соединения, выглядят следующим образом:

1. Клиент устанавливает соединение, используя имя сервера (или псевдоним с тем же именем).
2. Клиент проверяет наличие соединения с IndustrialSQL Server 7.1. Если IndustrialSQL Server, с которым устанавливается соединение, более ранней версии, то появится сообщение об ошибке.
3. Клиент считывает значение системного параметра *InSQLListenOn* из базы данных *Runtime*. Это `\\.\pipe\wwsql`.

---

**Примечание.** Параметр *InSQLListenOn* используется для сохранения информации соединения для клиентов, которым требуется доступ к базам данных IndustrialSQL Server. При запуске IndustrialSQL Server этот параметр обновляется до значения реестра *ListenOn*, хранящегося под ключом:

`HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\MSSQLServer\InSQL`

---

4. Клиент проверяет реестр на предмет псевдонима протокола named pipes для указанного имени сервера (см. шаг 1).  
Если псевдоним присутствует, клиент считывает значение реестра *InSQL ListenOn* для этого псевдонима и считывает имя сервера из строки соединения.
5. Клиент "сходу" создает новую строку соединения, заменяя точку (.) в строке соединения либо на указанное имя сервера (см. шаг 1), либо на имя псевдонима named pipe (см. шаг 3). Например, `\\MyServer\pipe\wwsql`.
6. Клиент отсоединяется от сервера.

Клиент вновь устанавливает соединение, используя новую строку соединения.

## Запрет автоматического создания строки соединения

Может возникнуть надобность в запрете автоматического создания строки соединения в следующих ситуациях:

1. Вы знаете, как создавать и управлять строками соединения и предпочтете создавать постоянные псевдонимы с соответствующей строкой соединения с помощью утилиты SQL Server Client Network.
2. Вы хотите ускорить процесс соединения клиента.
3. Вы хотите увеличить скорость выборки данных приложением-клиентом. Строка соединения автоматически создается для протокола named pipes, который является уровнем межпроцессового общения, стоящим выше сетевого протокола. Например, если вы используете TCP/IP, то соединение через новую строку будет происходить через протокол named pipes, что гораздо медленнее.

Чтобы запретить автоматическое создание строки соединения для конкретного клиента, установите значение ключа реестра

**SmartConnectionEnabled** для этого клиента на 0. Например, для InSQL Trend этот ключ будет находиться в:

```
HKEY_CURRENT_USER
\Software
  \Wonderware
    \InSQL
      \Clients
        \Trend
          \7.1
            \Connect
```

## Клиенты OLE DB IndustrialSQL Server 7.1

К клиентам OLE DB IndustrialSQL Server относятся: InSQL Report, ActiveDataGrid, ActiveGraph, ActiveTagBrowser, ActiveTimeSelector и поставщик архивов IndustrialSQL Server для InTouch. Для обращения к данным эти клиенты используют поставщика архивов InSQL OLE DB. Для использования поставщика архивов InSQL OLE DB необходимо напрямую соединиться с Microsoft SQL Server. Имя сервера Microsoft SQL Server обычно представлено именем компьютера, на котором он запущен.

---

**Примечание.** Если вы обращаетесь к серверу Internet через InSQL Report или через элементы InSQL ActiveX, для каждого запроса к данным создается новое соединение.

---

## Правила настройки клиента ODBC

В дополнение к настройке сервера библиотеки БД должна быть проведена настройка ODBC для приложений-клиентов ODBC, которые обращаются к IndustrialSQL Server по сети. К клиентам ODBC относятся программы Microsoft Access, Microsoft Excel и Microsoft Word.

Настройка источника данных для клиента ODBC требует указания имени сервера библиотеки БД. Также можно настроить сетевой адрес источника данных, сетевую библиотеку и стандартное имя базы данных.

Можно устанавливать соединение клиента/сервера ODBC только для IndustrialSQL Servers, настроенного на использование 32-битной библиотеки БД. Приложения-клиенты ODBC используют те же стандартные сетевые библиотеки и общие параметры, что и библиотеки БД, если эти установки не изменены при настройке источника данных.

---

**Примечание.** Если вы настроили соединение ODBC для использования с IndustrialSQL Server 7.0 и хотите теперь использовать его же с IndustrialSQL Server 7.1, следует изменить строку соединения для ODBC с `\\.\pipe\sql\query` на `\\.\pipe\wwsql`.

---

Подробнее о настройке ODBC см. в *Руководстве администратора системы FactorySuite*.

# Управление безопасностью

В этой главе объясняются основы управления безопасностью IndustrialSQL Server. Поскольку в систему IndustrialSQL Server встроена система Microsoft SQL Server, IndustrialSQL Server пользуется возможностями безопасности Microsoft SQL Server.

После установки IndustrialSQL Server создаются стандартные учетные записи, которые можно использовать для доступа к системе приложений-клиентов. Стандартные учетные записи предоставляют готовые функции, позволяющие не создавать учетные записи для начала использования системы.

Подробнее о функциях безопасности, описанных в этой главе, см. в документации Microsoft SQL Server.

## Содержание

- Уровни безопасности
- Инструменты безопасности
- Проверка режима безопасности
- Управление идентификаторами пользователей сервера SQL
- Управление пользователями и группами
- Управление правами допуска
- Управление паролями
- Изменение идентификатора пользователя wwAdmin

## Уровни безопасности

Задача безопасности, которую решает Microsoft SQL Server, это управление правами доступа к конкретным объектам базы данных и правами на изменение этих объектов. Например, безопасность Microsoft SQL Server может использоваться для управления правами записи в таблицу *Tag*. Для получения доступа к объектам базы данных пользователь должен пройти через несколько уровней безопасности.

### Уровень 1: безопасность операционной системы

Первое, что должен сделать пользователь, это зарегистрироваться в операционной системе на своем компьютере. В Windows NT для регистрации требуется ввод правильной учетной записи пользователя, состоящей из имени пользователя и пароля. Учетная запись Windows NT может также использоваться для доступа к сетевым ресурсам домена.

### Уровень 2: безопасность Microsoft SQL Server

Параметры безопасности Microsoft SQL Server определяют, кто имеет доступ к приложению Microsoft SQL Server. Доступ предоставляется пользователю, который имеет на сервере идентификатор пользователя. Под этим идентификатором пользователь известен системе Microsoft SQL Server. Идентификаторы пользователей хранятся в таблице *syslogins*. Существует два способа управления доступом на уровне сервера, как показано в следующей таблице:

Режим безопасности	Описание
Режим аутентификации Windows NT	Для доступа к Microsoft SQL Server пользователи могут использовать имя пользователя и пароль Windows NT. Отдельная учетная запись для Microsoft SQL Server не обязательна. Microsoft SQL Server для проверки регистрации использует механизм безопасности Windows NT. Интегрированная безопасность, тем не менее, может использоваться только при надежном соединении (через multi-protocol или named pipes).
Смешанный режим	Если идентификатор пользователя совпадает с сетевым именем пользователя Windows NT, то проверка проводится механизмом безопасности Windows NT. Если же совпадения имен пользователей нет, то используется механизм безопасности Microsoft SQL Server.

По умолчанию IndustrialSQL Server использует смешанный режим безопасности. Как настроить IndustrialSQL Server на режим аутентификации Windows NT, объясняется в документации по Microsoft SQL Server для NT.

## Уровень 3: безопасность базы данных

Параметры безопасности базы данных определяют, кто имеет доступ к базе данных Microsoft SQL Server. Перед установкой безопасности на уровне базы данных необходимо создать идентификатор пользователя в Microsoft SQL Server. Безопасность базы данных реализуется путем привязки идентификатора пользователя (безопасность на уровне сервера) с именем пользователя базы данных.

Имена пользователей хранятся в таблице *sysusers* в базе данных *master*. При попытке пользователя обратиться к базе данных Microsoft SQL Server просматривает запись в таблице *sysusers* и затем ищет соответствие в таблице *syslogins*. Если Microsoft SQL Server не находит имя пользователя, доступ не разрешается.

## Уровень 4: безопасность объектов и выражений

Типы действий, которые может предпринять пользователь над базой данных основываются на информации учетной записи. Эти разрешенные действия называются доступом. Существует два типа доступа: доступ к объекту и доступ к выражениям.

Допуск	Описание
Объект	Определяет действия, которые пользователь может проводить над существующими объектами базы данных. К объектам базы данных относятся таблицы, индексы, представления, стандарты, триггеры, правила и процедуры. Доступ к объекту предоставляется и отбирается владельцем (создателем) объекта.
Выражение	Определяет, кто может использовать конкретные выражения Transact-SQL. Выражения базы данных включают в себя такие команды как SELECT, INSERT или DELETE. Доступ к выражениям, называемый также командным доступом, может быть предоставлен или отобран только администратором системы или владельцем базы данных.

## Группы

Для облегчения работы с большим количеством пользователей базы данных каждое имя пользователя может быть включено в группу пользователей Microsoft SQL Server. Все члены группы получают тот доступ, который разрешен этой группе. Например, если пользователь "МэриБ" добавлен в группу "Опытные пользователи", то этот пользователь автоматически получает соответствующий этой группе доступ.

Добавляя новую группу, вы сначала должны дать IndustrialSQL Server права доступа к этой группе, которые расположены в столбце *AccessLevel* таблицы *UserDetail*. Новые пользователи и группы автоматически добавляются в таблицу *UserDetail* каждый час (благодаря тому, что запущен SQL Agent). Их можно добавить и вручную при помощи процедуры **ww\_UserDetailUpdate**.

В группы можно добавлять пользователей и группы Windows NT, а также пользователей и группы Microsoft SQL Server.

## Инструменты безопасности

Для управления безопасностью можно пользоваться программой SQL Server Enterprise Manager. С ее помощью можно:

- Установить безопасность сервера на режим аутентификации Windows NT или на смешанный режим.
- Управлять идентификаторами пользователей Сервера SQL.
- Управлять пользователями базы данных.
- Управлять допуском к объектам.
- Управлять допуском к выражениям.

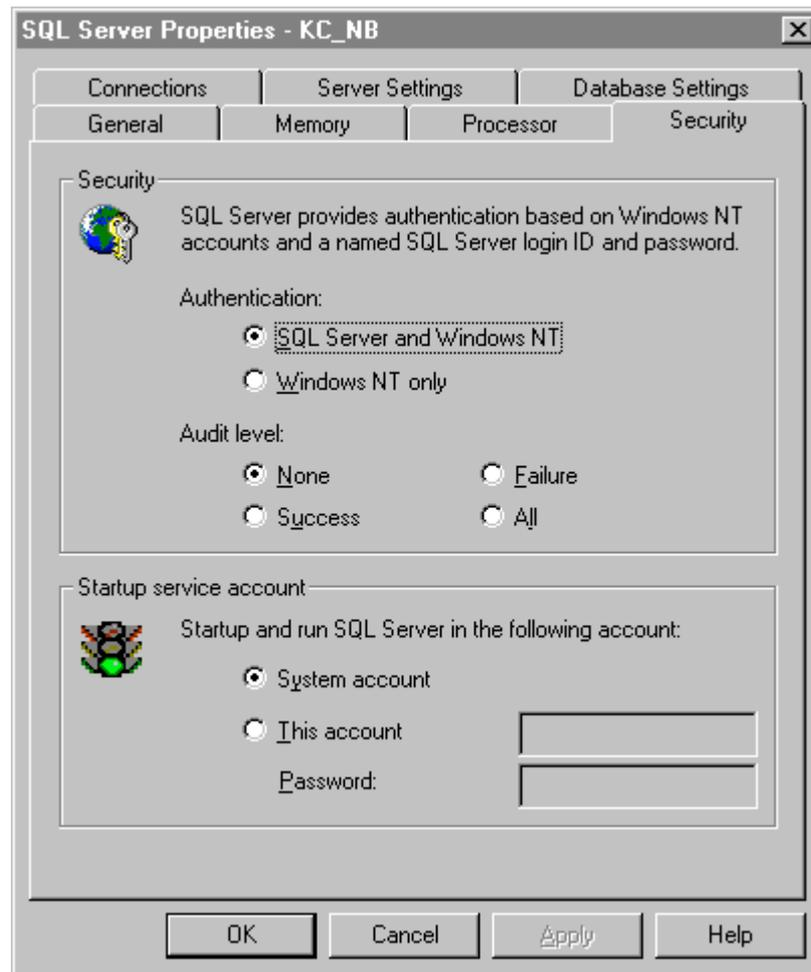
Если IndustrialSQL Server установлен для работы с существующей системой Microsoft SQL Server, настроенной на режим аутентификации Windows NT или смешанный режим, используйте Microsoft SQL Security Manager для предоставления группам и пользователям Windows NT доступа к Microsoft SQL Server.

## Проверка режима безопасности

По умолчанию IndustrialSQL Server использует смешанный режим безопасности. Режим безопасности настраивается при помощи SQL Server Enterprise Manager.

➤ **Как проверить режим безопасности:**

1. В дереве консоли нажмите правой кнопкой мыши на сервер.
2. В контекстном меню выберите **Properties**.
3. Появится диалоговое окно **SQL Server Properties**.
4. Выберите панель **Security**.



5. Проверьте режим безопасности. Подробнее об использовании в IndustrialSQL Server режима аутентификации Windows NT см. в документации по Microsoft SQL Server для NT.
6. Нажмите **OK**.

## Управление идентификаторами пользователей сервера SQL

Для того, чтобы пользователь получил доступ к базе данных, его идентификатор должен быть занесен в Microsoft SQL Server. Создавать идентификаторы пользователей на сервере SQL может только системный администратор (Sa). Управление идентификаторами пользователей осуществляется посредством SQL Server Enterprise Manager.

---

**Примечание.** Создание индивидуальной учетной записи не обязательно для каждого пользователя, если сервер SQL работает в режиме аутентификации Windows NT. Имена пользователей на основе Windows NT можно автоматически переключать на имена пользователей сервера SQL при помощи SQL Security Manager.

---

## Стандартные идентификаторы пользователей IndustrialSQL Server

В IndustrialSQL Server заранее настроены следующие идентификаторы пользователей:

Идентификатор пользователя	Описание
wwUser	Пользователь с правом "только чтение", который имеет доступ ко всем данным, но не может изменять их или оказывать влияние на ресурсы базы данных.
wwPower	Пользователю разрешается просмотр данных, создание объектов и внесение изменений в неосновные таблицы.
wwAdmin	Пользователь, имеющий право на доступ и изменение любых данных и создание объектов.  Не имеет права изменять свойства таблицы и уничтожать базу данных.
wwdbo	Владелец базы данных. Полные полномочия.

Стандартной базой данных для каждой из этих учетных записей является база данных *Runtime* IndustrialSQL Server. Эта модель стандартных настроек безопасности является отправной точкой для последующей настройки системной безопасности и подходит для многих конфигураций. Эта модель может быть позже изменена системным администратором.

➤ **Как просмотреть всех пользователей сервера SQL:**

1. В дереве консоли SQL Server Enterprise Manager разверните дерево сервера SQL.
2. Откройте папку **Security**.

3. Откройте папку **Logins**.
4. Все стандартные учетные записи пользователей сервера SQL появятся в окне справа от дерева консоли.
5. Подробнее см. документацию Microsoft.

## Добавление идентификатора пользователя

Системный администратор (sa) может добавлять конкретные параметры идентификатора пользователя, такие как пароль, имя пользователя, стандартную базу данных и стандартный язык. Если пользователь не получает имени пользователя в стандартной базе данных, то используется его идентификатор.

Если большое количество пользователей соединяется с базой данных на равных правах допуска, то создание единого идентификатора и имени пользователя для допуска всех этих пользователей уменьшит объем работы по управлению учетными записями.

### ➤ Как добавить идентификатор пользователя на сервере SQL:

1. В дереве консоли SQL Server Enterprise Manager разверните дерево сервера, в который нужно добавить идентификатор пользователя.
2. Откройте папку **Security**.
3. В дереве консоли нажмите правой кнопкой мыши на папке **Logins**.
4. В контекстном меню выберите **Properties**.

Появится диалоговое окно **SQL Server Login Properties**.

The screenshot shows the 'SQL Server Login Properties - New Login' dialog box. It has three tabs: 'General', 'Server Roles', and 'Database Access'. The 'General' tab is active. It contains the following fields and options:

- Name:** A text box containing 'wwSuperUser'.
- Authentication:** A section with three radio buttons:
  - Windows NT authentication. Below it is a 'Domain:' dropdown menu and a 'Security access:' section with two radio buttons:  Grant access and  Deny access.
  - SQL Server authentication. Below it is a 'Password:' text box filled with asterisks.
- Defaults:** A section with a database icon and the text 'Specify the default language and database for this login.' Below it are two dropdown menus:
  - Database:** Set to 'Runtime'.
  - Language:** Set to '<Default>'.

At the bottom of the dialog are three buttons: 'OK', 'Cancel', and 'Help'.

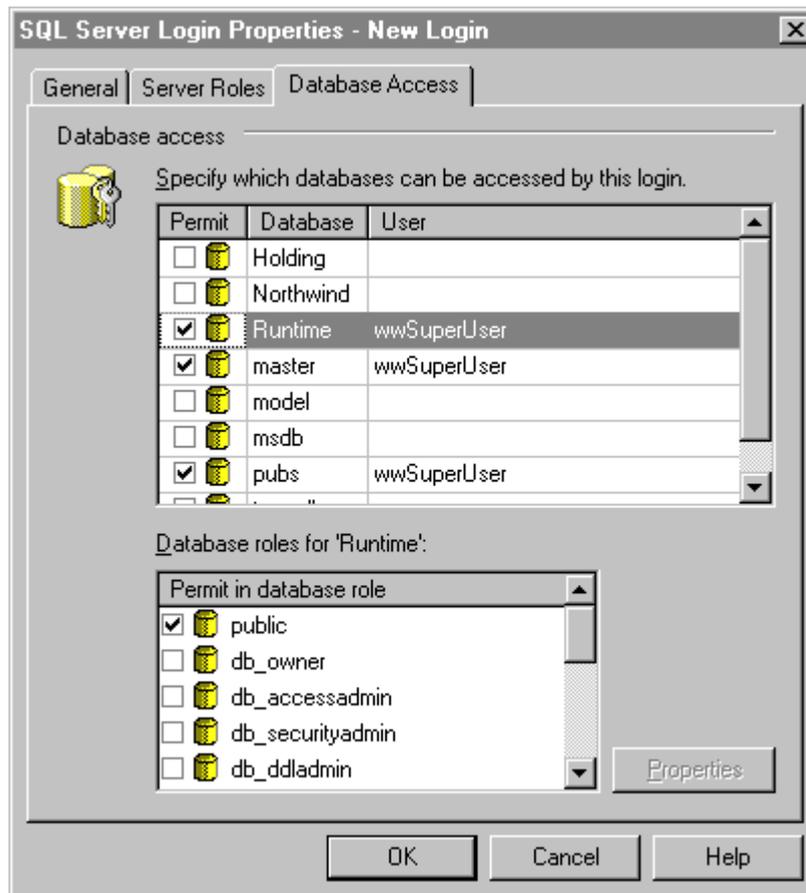
5. В поле **Name** введите имя нового идентификатора пользователя.
6. В группе параметров **Authentication** настройте новый идентификатор пользователя на режим аутентификации Windows NT или смешанный режим.

Если используется смешанный режим, то следует ввести пароль для идентификатора пользователя.

7. В списке **Database** выберите базу данных, которую идентификатор пользователя будет использовать по умолчанию.
8. Выберите язык из списка **Language** или оставьте <Default> для использования языка Английский (США).
9. Откройте панель **Server Roles**.



10. Для помещения нового пользователя в существующую группу(ы) пометьте соответствующие флажки в списке.
11. Откройте панель **Database Access**.



12. Столбец **User** содержит имя пользователя, используемое для идентификатора пользователя. Стандартное имя пользователя то же, что и его идентификатор.

13. Выберите базы данных, доступ к которым возможен через новый идентификатор пользователя.

После выбора базы данных доступные группы для этой базы данных появятся в списке **Database Roles**.

14. По умолчанию все новые идентификаторы пользователей являются членами группы базы данных **Public**. Вы можете выбрать дополнительную или иную группу конкретной базы данных.

15. Закончив настройку, нажмите **OK**.

16. Если вы создали идентификатор пользователя с паролем сервера SQL, то будет предложено подтвердить новый пароль. Повторно введите пароль и нажмите **OK**.

## Управление пользователями и группами

Прежде чем пользователь получает доступ к базе данных, его идентификатор и имя пользователя должны быть добавлены в базу данных сервера SQL. Только системный администратор (sa) может добавить идентификатор пользователя в сервер SQL. Также только системный администратор может добавить пользователя в группу. Если имя группы не указано, пользователь добавляется только в общую группу, которая включает в себя всех пользователей.

### Стандартные пользователи и группы IndustrialSQL Server

Стандартные пользователи IndustrialSQL Server уже имеют пароли и принадлежат к группе базы данных. Исходные пароли соответствуют именам пользователей. Следующая таблица описывает имена пользователей IndustrialSQL Server, группу базы данных, к которой они принадлежат, и действия, которые они могут совершать по отношению к базе данных  
*Runtime:*

ID	Имя	Группа БД	Допуск
wwUser	wwUser	wwUsers	SELECT во всех таблицах. INSERT, UPDATE, DELETE в таблицах <i>GroupTagList, TagGroup, Annotation</i>
wwPower	wwPower	wwPowerUsers	CREATE таблицу CREATE представление CREATE хранимую процедуру CREATE стандарт CREATE правило SELECT во всех таблицах INSERT, UPDATE, DELETE в групповых таблицах
wwAdmin	wwAdmin	wwAdministrators	CREATE таблицу CREATE представление CREATE хранимую процедуру CREATE стандарт CREATE правило DUMP базу данных DUMP транзакцию SELECT, INSERT, UPDATE, DELETE во всех таблицах
wwdbo	dbo	Public	Полные возможности владельца базы данных.

Если вы изменяете имя пользователя или пароль для идентификатора **wwAdmin**, то следует также изменить имя пользователя или пароль в реестре при помощи программы InSQL Control.

Подробнее см. в разделе "Изменение идентификатора пользователя **wwAdmin**" далее в этой главе.

➤ **Как просмотреть всех пользователей и группы базы данных:**

1. В дереве консоли SQL Enterprise Manager разверните дерево сервера SQL.
2. Разверните папку **Databases**.
3. Разверните базу данных, для которой вы хотите просмотреть всех пользователей и группы.



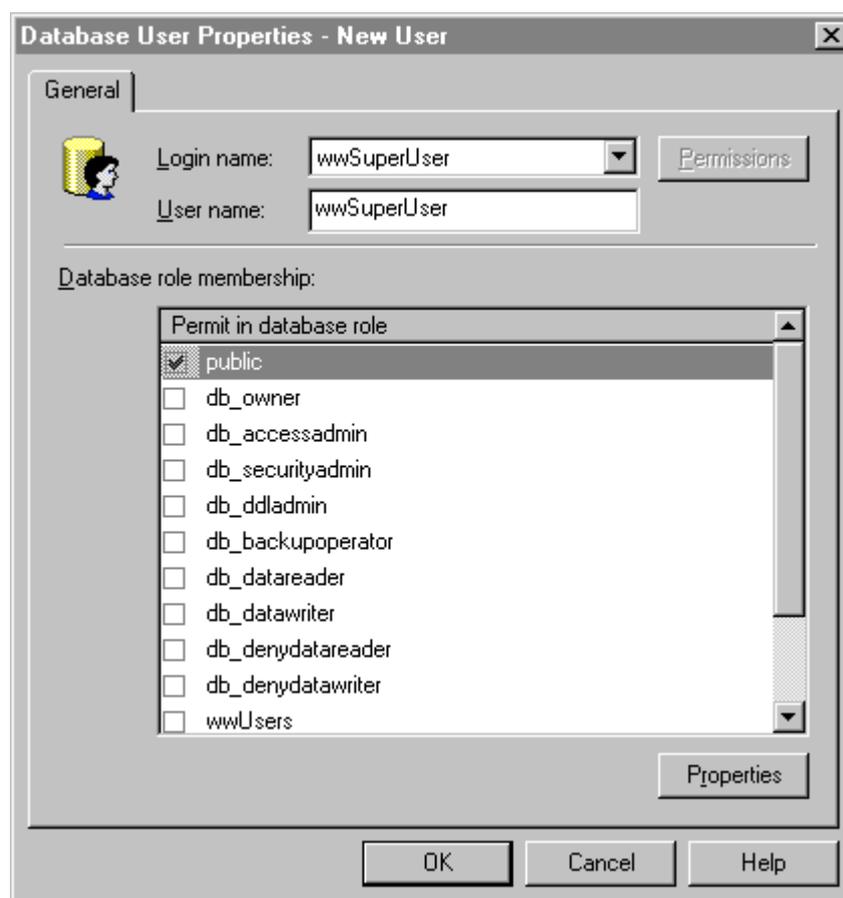
4. Для просмотра всех пользователей выберите папку **Users**. Все пользователи будут показаны в окне справа от дерева консоли.
5. Для просмотра всех групп выберите папку **Roles**. Все группы будут показаны в окне справа от дерева консоли.

## Добавление пользователя

Программа SQL Enterprise Manager позволяет легко добавлять пользователей базы данных.

### ➤ Как добавить пользователя базы данных:

1. В дереве консоли разверните сервер и базу данных, для которых вы хотите добавить нового пользователя. Например, база данных *Runtime*.
2. Нажмите правой кнопкой мыши на папке **Users**. В появившемся меню выберите **New Database User**.
3. Появится диалоговое окно **Database User Properties**.



4. В списке **Login name** выберите идентификатор для имени пользователя. (Идентификатор пользователя должен быть создан до присвоения имени пользователя БД).
5. В поле **User name** введите новое имя пользователя.
6. В окне **Database role membership** выберите группу, членом которой будет этот пользователь.
7. Нажмите **OK**.

## Добавление новой группы базы данных

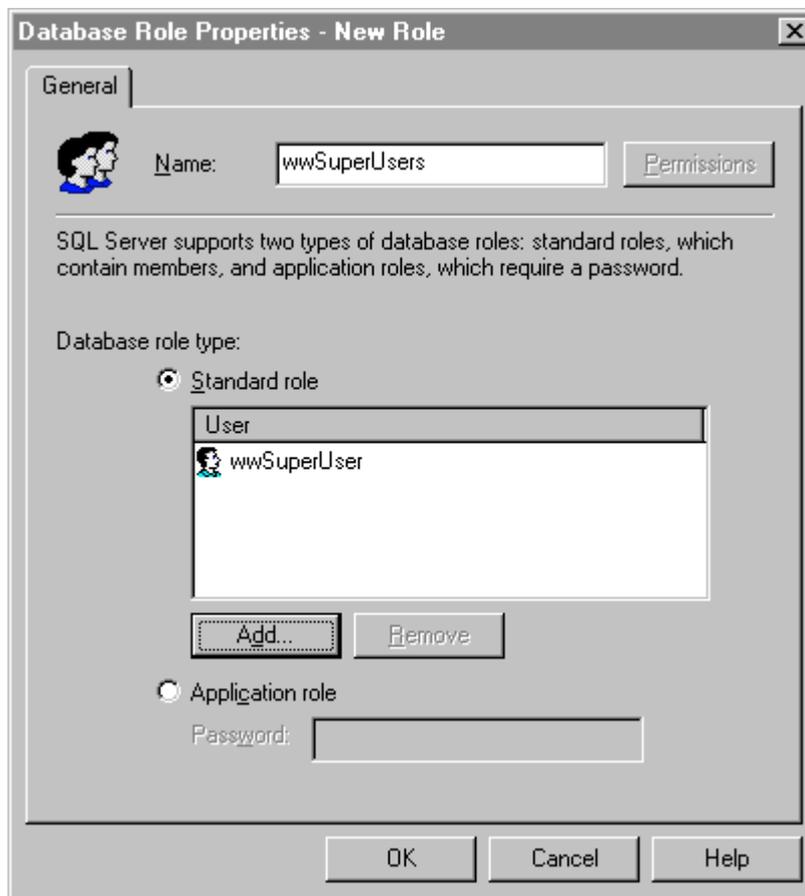
Использование групп облегчает управление правами доступа для большого количества пользователей. Вы можете создать группу, установить права доступа для этой группы и добавить в нее пользователей. Все пользователи обладают правами допуска той группы, к которой они принадлежат.

Пользователь базы данных может принадлежать к любому количеству групп.

### ➤ Как создать группу:

1. В дереве консоли разверните сервер и базу данных, для которых вы хотите добавить новую группу. Например, база данных *Runtime*.
2. Нажмите правой кнопкой мыши на папке **Roles**. В появившемся меню выберите **New Database Role**.

Появится диалоговое окно **Database Role Properties**.



3. В поле **Name** введите имя новой группы.
4. Для добавления новой группы нажмите **Add**. В появившемся диалоговом окне выберите пользователя из списка и нажмите **OK**.
5. Нажмите **OK**.

## Управление правами допуска

Права допуска — это действия, которые пользователю позволено совершать в базе данных сервера SQL. Вы можете дать права на доступ к объектам или выражениям любому пользователю группы базы данных. Пользователи обладают правами допуска тех групп, к которым они принадлежат.

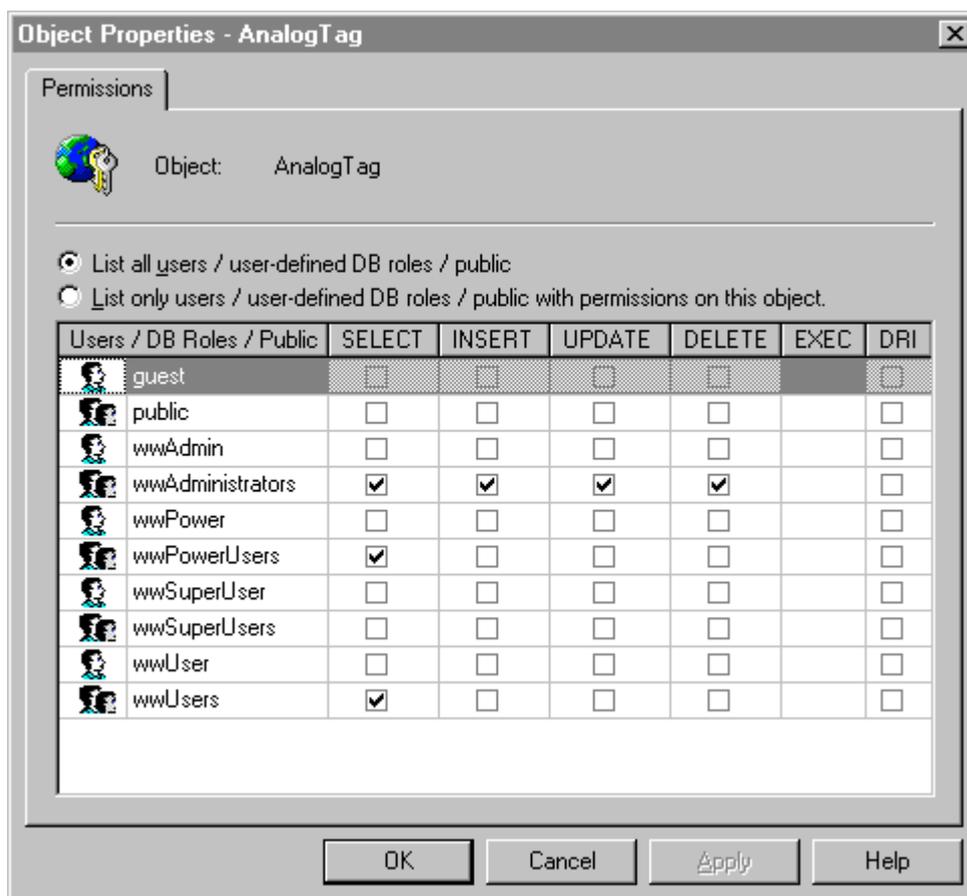
### Установка прав допуска к объекту

Права допуска к объекту определяют действия, которые пользователь может проводить над объектами базы данных, как то таблицы, индексы, представления, стандарты, триггеры, правила и процедуры. Для того, чтобы давать и отбирать права доступа, вы должны быть владельцем (создателем) объекта. Права доступа к объекту можно предоставить пользователю, группе или указать права допуска к объекту.

➤ **Как предоставить права допуска к объекту:**

1. В дереве консоли разверните сервер и базу данных, к объекту которых вы собираетесь дать доступ.
2. Выберите в базе данных тип объекта (таблица, хранимая процедура, представление и т.п.).
3. В панели деталей нажмите правой кнопкой мыши на объекте. В появившемся меню выберите **All Tasks** и затем нажмите **Manage Permissions**.

Появится диалоговое окно **Object Properties**.

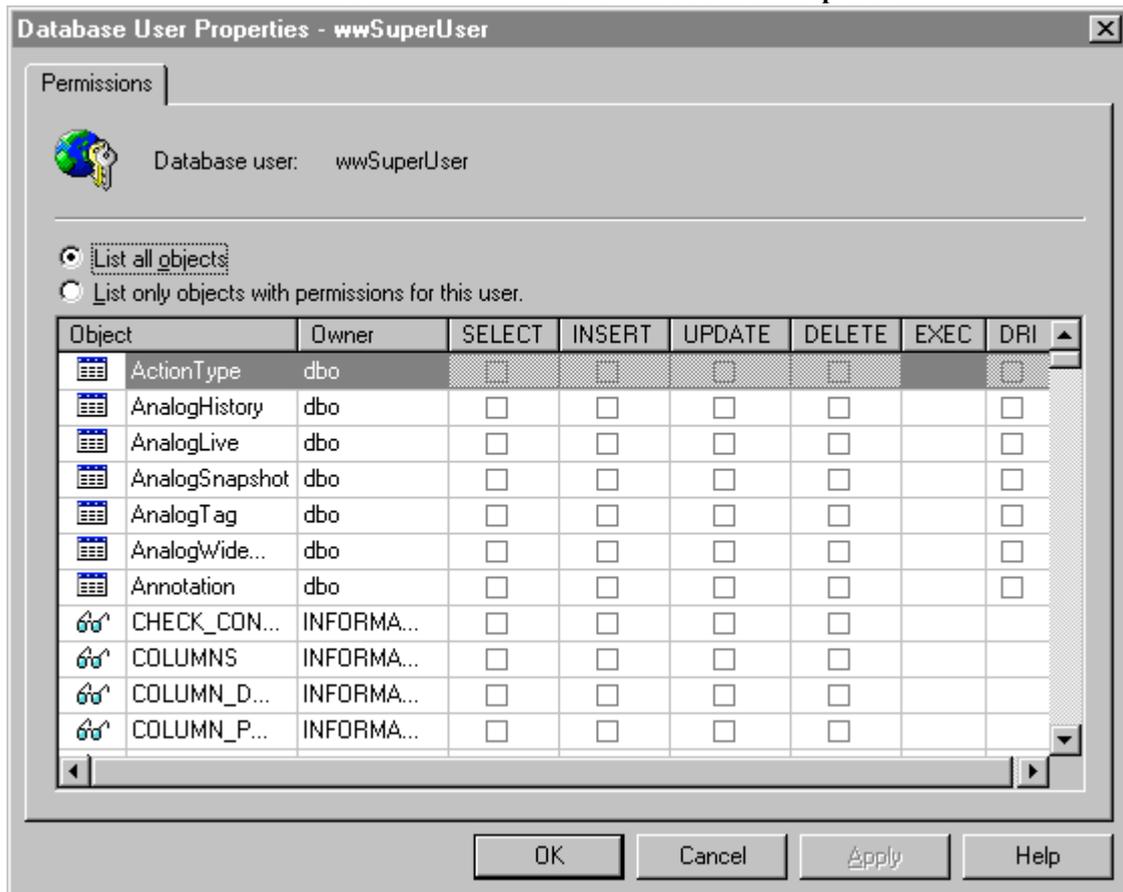


4. Для каждого пользователя группы пометьте права доступа к объекту.
5. Нажмите **OK**.

➤ **Как предоставить права доступа к объекту пользователю или группе:**

1. В дереве консоли разверните сервер и базу данных, к объекту которых вы собираетесь дать доступ.
2. Разверните папку **Users** или **Roles**.
3. В окне деталей нажмите правой кнопкой мыши на пользователя/группу. В появившемся меню выберите **All Tasks** и затем нажмите **Manage Permissions**.

Появится диалоговое окно **Database User/Role Properties**.



4. Выберите объект, для которого вы будете определять права доступа, а затем пометьте права, которые будут предоставлены для объекта.
5. Нажмите **OK**.

## Установка прав доступа к выражениям

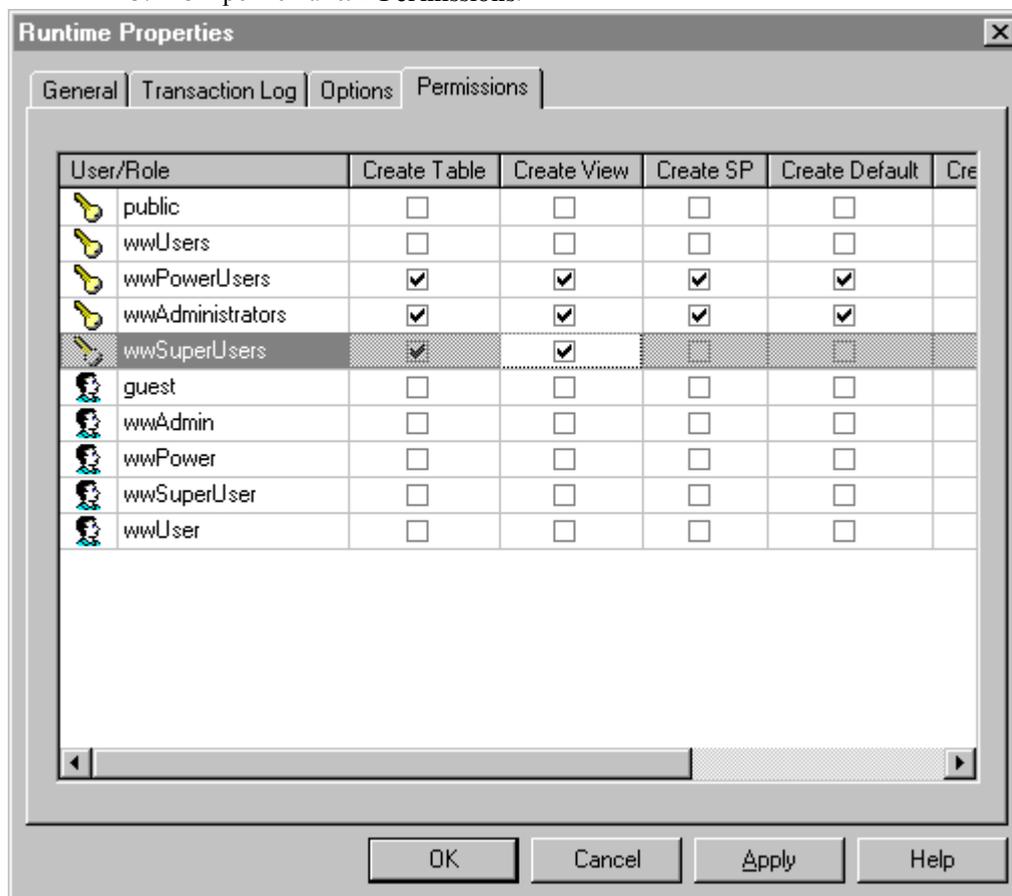
Права доступа к выражениям определяют, кто имеет право на использование выражений Transact-SQL, таких как SELECT, INSERT или DELETE. Для того, чтобы предоставлять или отбирать права доступа к выражениям, вы должны быть системным администратором или владельцем базы данных.

### ➤ Как установить права доступа к выражениям:

1. В дереве консоли разверните сервер и базу данных, для которых вы хотите определить права доступа к выражениям.
2. Нажмите правой кнопкой мыши на папку базы данных. В контекстном меню выберите **Properties**.

Появится диалоговое окно **Database Role Properties**.

3. Откройте панель **Permissions**.



**Примечание.** Допуск к выражению CREATE DATABASE может быть установлен только из базы данных master.

4. Выберите пользователя или группу, для которой вы будете определять права доступа, а затем пометьте права, которые будут определены для пользователя или группы.
5. Нажмите **ОК**.

## Управление паролями

Стандартные пароли для всех готовых учетных записей IndustrialSQL Server совпадают с именами пользователей. Например, пароль для пользователя "wwUser" будет "wwUser."

---

**Внимание!** Системному администратору очень важно иметь личный пароль для Microsoft SQL Server. Если нет пароля системного администратора, Wonderware сохраняет за собой право на отказ в технической поддержке.

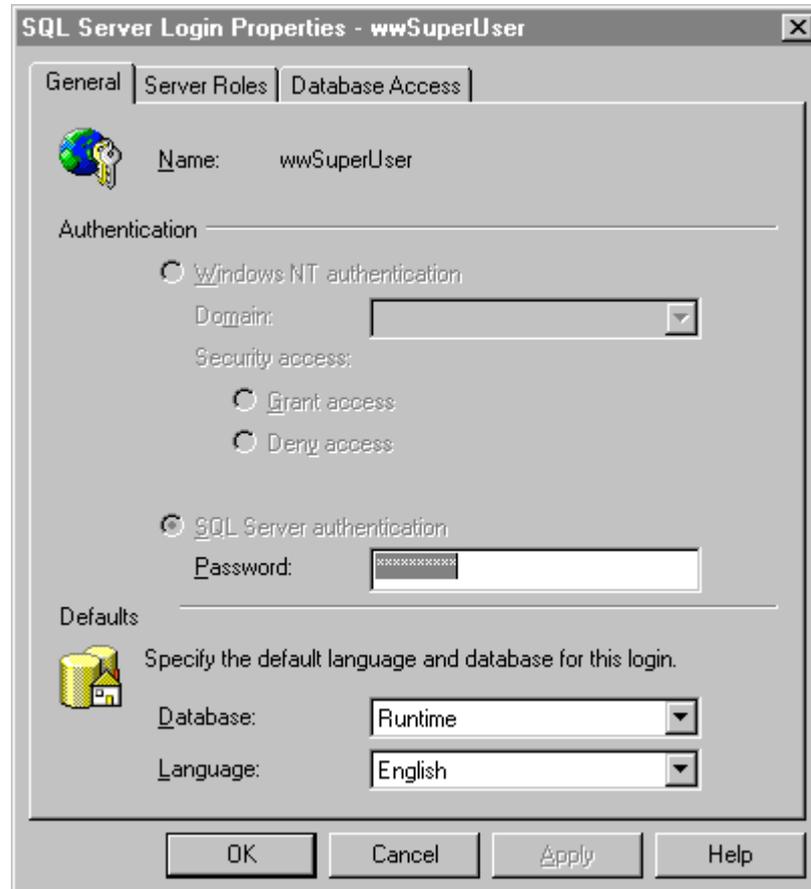
---

Если вы являетесь системным администратором, вы можете изменить пароль для любой учетной записи. Если вы не системный администратор, то вы можете изменить только свой пароль.

➤ **Как сменить пароль:**

1. В дереве консоли разверните сервер, на котором хотите изменить пароль.
2. Откройте папку **Security**.
3. Выберите папку **Logins**.
4. В панели деталей нажмите правой кнопкой мыши на имени пользователя, пароль которого хотите изменить. В меню выберите **Properties**.

Появится диалоговое окно **SQL Server Login Properties**.



5. В поле **Password** введите новый пароль.
6. Нажмите **ОК**.
7. В появившемся диалоговом окне подтвердите новый пароль и нажмите **ОК**.

## Изменение идентификатора пользователя wwAdmin

При загрузке системы программа InSQL Control (INSQLCNTL.EXE) просматривает реестр для определения имени пользователя и пароля, с которыми будет загружен Microsoft SQL Server.

Если вы измените имя пользователя или пароль "wwAdmin" без соответствующих изменений идентификатора пользователя "wwAdmin" в реестре, InSQL Control не сможет зарегистрироваться на Microsoft SQL Server, и, соответственно, система не запустится.

Идентификатор пользователя "wwAdmin" можно изменить из InSQL Control (INSQLCNTL.EXE).

Подробнее см. в главе 5, "InSQL Control".

# Регистрация ошибок

В состав операционной системы Windows NT входят утилиты для мониторинга производительности системы и приложений. В дополнение к этому, система IndustrialSQL Server регистрирует ошибки в специальных файлах на диске. Функция регистрации ошибок IndustrialSQL Server управляется с помощью установок в реестре. Эти установки определяют тип регистрируемой информации, а также место ее регистрации.

В данной главе описываются методы регистрации ошибок системы IndustrialSQL Server, включая расшифровку сообщений об этих ошибках. Кроме того, здесь можно найти расшифровку ошибок, выдаваемых в процессе работы драйвера ввода/вывода.

## Содержание

- Управление регистрацией ошибок
- Категории ошибок
- Настройка места регистрации ошибок
- Настройка журнала событий Windows NT
- Расположение файлов регистрации ошибок
- Сообщения об ошибках драйвера ввода/вывода

## Управление регистрацией ошибок

Система IndustrialSQL Server состоит из нескольких программных уровней, обеспечивающих хранение и выборку данных. В процессе работы системы на разных уровнях могут возникать ошибки или ситуации, требующие их регистрации. Данный раздел описывает категории ошибок, регистрируемых системой IndustrialSQL Server, а также место их записи и параметры, управляющие регистрацией ошибок.

### Категории ошибок

Информация в журнале ошибок подразделяется на следующие категории:

Категория	Описание
FATAL	Фатальные ошибки. Продолжение обработки невозможно. Ошибки такого рода приводят к закрытию системы.
CRITICAL	Критические ошибки. Ошибки этого типа вызывают некорректное функционирование систем хранения и выборки данных, что может привести к потере или разрушению данных.
ERROR	Типичные ошибки. Например, ошибки определения адреса в процессе запуска системы. Ошибка данного вида могут привести к штатному закрытию системы, однако в большинстве случаев не будут препятствовать нормальной работе системы.
EL_WARNING	Предупреждения в действительности не являются ошибками, а представляют собой уведомления о параметрах настройки или о произошедших событиях. В качестве примера события, по которому выдается предупреждение, можно привести сбой привязки точки входа динамически связываемой процедуры к необязательной функции.
INFO/DISPLAY	Информационные сообщения, показывающие ход выполнения запуска или активные хранилища данных.
EL_DEBUG	Сообщения отладчика, обычно не появляющиеся в коммерческих версиях системы.

## Настройка места регистрации ошибок

Система IndustrialSQL Server в зависимости от настроек может хранить информацию об ошибках в следующих местах:

- Экран консоли
- Файл журнала
- Журнал событий Windows NT

Изменить места записи и вывода зарегистрированной информации можно только для информационных, предупреждающих и отладочных сообщений. Для типичных, критических и фатальных ошибок эта функция не поддерживается. Настройка мест хранения регистрационной информации для информационных, предупреждающих и отладочных сообщений производится в следующем ключе реестра:

```
HKEY_LOCAL_MACHINE
  \SOFTWARE
    \Wonderware
      \InSQL
        \Programs
```

Настройки для этих значений реестра представляют собой битовые маски:

- 1 (бит 0)    вывод на экран консоли
- 2 (бит 1)    вывод в файл журнала
- 4 (бит 2)    вывод в журнал событий Windows NT Event Viewer

Регистрацию сообщений EL\_ERROR, EL\_CRITICAL и EL\_FATAL настраивать нельзя. Для настройки регистрации сообщений типа EL\_DISPLAY, EL\_INFO, EL\_WARNING и EL\_DEBUG можно использовать любую маску или их комбинацию. Например, если требуется настроить вывод предупредительных сообщений в файл и на экран, значение EL\_WARNING в реестре должно быть равным 3 (установка бит 0 и бит 1).

Значения, присваиваемые по умолчанию, приведены в таблице ниже

Ключ	Значение по умолчанию
EL_DEBUG	0
EL_DISPLAY	0
EL_INFO	3
EL_WARNING	7

Для настройки регистрации ошибок по конкретной задаче системы IndustrialSQL Server можно добавить значения в подключе следующего ключа:

```
HKEY_LOCAL_MACHINE
  \SOFTWARE
    \Wonderware
      \InSQL
        \Programs
```

Представим, что требуется регистрировать ошибки типа EL\_ERROR, EL\_CRITICAL и EL\_FATAL в рамках всей системы IndustrialSQL Server, а сообщения EL\_INFO от подсистемы хранения аналоговых данных заносить в журнал событий Windows NT и в файл журнала ошибок. Тогда потребуется добавить значение EL\_INFO в следующий ключ и установить для него значение 6 (1 и 2 бит)

```
HKEY_LOCAL_MACHINE
  \SOFTWARE
    \Wonderware
      \InSQL
        \Programs
          \Storage
            \1
```

## Настройка журнала событий Windows NT

Чтобы корректно использовать журнал событий Windows NT (EVENTVWR.EXE) для регистрации ошибок системы IndustrialSQL Server, необходимо правильно настроить реестр. В указанном ниже ключе должен существовать подключ InSQL:

```
HKEY_LOCAL_MACHINE
  \System
    \CurrentControlSet
      \Services
        \EventLog
          \Application
```

Подключ IndustrialSQL Server должен содержать следующие значения:

Подключ	Описание
EventMessageFile: REG_EXPAND_SZ: <i>путь</i>	Содержит полный путь и название библиотеки обработчика ошибок для журнала событий - PDSMSG.DLL
TypesSupported: REG_DWORD: <i>битовая маска</i>	Битовая маска, задающая классы сообщений для регистрации в журнале событий. Битовые маски описаны в разделе "Настройка места регистрации ошибок" выше в данной главе.

## Просмотр зарегистрированной информации IndustrialSQL Server в журнале событий

После настройки реестра журнал событий Windows NT может быть использован для регистрации ошибок IndustrialSQL Server.

➤ **Как отыскать записи IndustrialSQL Server:**

1. В меню **Log** нажмите **Application**.
2. В меню **View** выберите **Filter Events**.

Появится диалоговое окно **Filter**.

3. В списке **Source** выберите **InSQL**.
4. Нажмите **OK**.

## Расположение файлов регистрации ошибок

Информацию по ошибкам можно регистрировать в одном или нескольких местах. Одним из таких мест является файл журнала на диске. Данный раздел содержит описание точного месторасположения таких файлов и их назначение.

Существует два типа журналов ошибок.

- Общий журнал запуска системы IndustrialSQL Server.
- Статус и запуск системы хранения IndustrialSQL Server.

## Запуск системы и статус подсистемы сервера

Файл журнала для регистрации статуса подсистемы сервера и запуска системы называется ERR\_ODS.ERR и располагается на диске в корне каталога циклических данных. Это каталог, в котором создаются архивные блоки. Найти его можно в значении *DynData* следующего ключа реестра:

```
HKEY_LOCAL_MACHINE
  \SOFTWARE
    \Wonderware
      \InSQL
        \Datapaths
```

## Запуск и статус системы хранения

Файл журнала для регистрации информации по запуску и статусу системы хранения находится в подкаталогах каждого архивного блока, находящихся в корне каталога циклических данных. Этот журнал содержит информацию об ошибках, возникающих при создании конкретного архивного блока. Всякий раз при создании нового блока создается новый файл журнала. Имя создаваемого файла журнала совпадает с именем архивного блока, но имеет расширение ERR. Файлы журналов ошибок хранятся в папке соответствующего архивного блока.

Более подробную информацию об архивных блоках можно найти в главе 4, "Управление системой хранения".

## Сообщения об ошибках драйвера ввода/вывода

Драйвер ввода-вывода (IODRIVER.EXE) выводит сообщения об ошибках в файл журнала ошибок. Расшифровка этих сообщений приводятся в следующей таблице.

**Примечание.** При установке ключа реестра debug генерируются также сообщения отладчика типа DEBUG.

№	Сообщение об ошибке	Тип	Описание	Комментарии
1	Setting Status (получение статуса)	Отладка	Ошибку вернул comStatus.UpdateStatus().	От FS1000.
2	Could not Create RTServer Thread (не удается создать нить RTServer)	Фатальная	Будет выполнен выход из приложения	
3	Error reading from registry (ошибка чтения реестра)	Фатальная	Будет выполнен выход из приложения	От FS1000.
4	Failure reading active image and/or database. (сбой чтения активного образа и/или базы данных)	Ошибка	Будет выполнен выход из приложения	От FS1000.
5	Reading analog configuration data. (считывание конфигурации аналоговых данных)	Отладка	Чтение из активного образа.	От FS1000.
5	Reading digital configuration data. (считывание конфигурации дискретных данных)	Отладка	Чтение из активного образа.	От FS1000.
5	Reading string configuration data. (считывание конфигурации текстовых данных)	Отладка	Чтение из активного образа.	От FS1000.
6	Error reading Active Image. (ошибка чтения активного образа)	Ошибка	Будет выполнен выход из приложения	От FS1000.
7	Ошибка проверки элемента: //node/app tag!item	Критическая	Игнорируется.	От FS1000.
8	Error logging on to <Имя сервера SQL> (ошибка входа на сервер SQL)	Фатальная	Будет выполнен выход из приложения	От FS1000.
9	Error Opening registry Key. (ошибка открытия ключа реестра)	Фатальная	Будет выполнен выход из приложения	От FS1000.

*(продолжение следует)*

*(продолжение)*

№	Сообщение об ошибке	Тип	Описание	Комментарии
10	Error opening registry key. (ошибка открытия ключа реестра)	Ошибка	Динамическое уведомление о том, что произведенные в реестре изменения работать не будут.	
11	Error setting registry notification. (ошибка установки реестра)	Ошибка	Динамическое уведомление о том, что произведенные в реестре изменения работать не будут.	
12	Invalid connection state transition from <state> on <event>. (недопустимый переход состояний подключения из <состояния 1> в <состояние 2>)	Критическая	Подключение восстановить невозможно. Это означает, что у пользователя нет возможности запустить/остановить процесс. Необходимо остановить/запустить IODRIVER.EXE.	
13	Connecting to node.app.topic failed (wwCliConnect() returns 0) (сбой подключения к узлу.приложению.теме (wwCliConnect() вернул 0))	Ошибка	Сбой установки соединения.	Выводит только каждую 100 <sup>-10</sup> ошибку для каждого цикла из idle->idle
14	Disconnecting from node.app.topic (отключение от узла.приложения.темы)	Информационная	Тема отключен.	Выводит только каждую 100 <sup>-10</sup> ошибку для каждого цикла из idle->idle
15	Сбой создания таймера	Критическая	Ни один из таймаутов не будет работать для всех подключений.	
16	Throttling: server node.app.topic is folded-on	Информационная		
17	Throttling: server node.app.topic is folded-off	Информационная		
18	Advising Data Item Tagname on topic node.app.topic failed.	Ошибка	Сбой функции advise	
19	Registering Data Item Tagname on topic node.app.topic failed.	Ошибка	Сбой регистрации.	
20	<nn> Errors during Advise process	Ошибка	Общее число ошибок в процессе advise (по любой причине).	
21	Unsupported data type in topic: (Неподдерживаемый тип данных в теме:) node.app.topic (узел.приложение.тема)	Ошибка	Тэг не является ни аналоговым, ни дискретным, ни текстовым.	

*(продолжение следует)*

*(продолжение)*

№	Сообщение об ошибке	Тип	Описание	Комментарии
22	"node.app.topic" Connected ("узел.приложение.тема" подключено)	Информационная	Текущая тема подключена в данный момент.	
23	<nn> "Advise Ack" events received for «node.app.topic» (События advise получены с "узла.приложения.темы")	Ошибка	Число обратных вызовов об отключении получено.	Выводит только каждую 100 <sup>10</sup> ошибку для каждого цикла из idle->idle
24	<nn> "Advise Ack" events received for «node.app.topic» (События advise ack получены с "узла.приложения.темы")	Отладка	Число обратных вызовов advise acknowledgments получено.	Выводит только каждую 100 <sup>10</sup> ошибку для каждого цикла из idle->idle
25	<nn> "Advise Nack" events received for "node.app.topic" (События advise Nack получены с "узла.приложения.темы")	Информационная	Число обратных вызовов негативных подтверждений advise.	Выводит только каждую 100 <sup>10</sup> ошибку для каждого цикла из idle->idle
26	node.app.topic Receiving Data (прием данных узла.приложения.темы)	Информационная		Единожды при первом получении данных (из FS1000).
27	Event Callback: (обратный вызов по событию) client overload for node.app.topic (клиент для узла.приложения.темы перегружен)	Предупреждение		Выводит только каждую 100 <sup>10</sup> ошибку для каждого цикла из idle->idle
28	IO Protocol overload on node.app.topic (possible data loss) (Протокол ввода-вывода в узле.приложении.теме перегружен (возможна потеря данных))	Предупреждение		Выводит только каждую 100 <sup>10</sup> ошибку для каждого цикла из idle->idle Ошибки 27 и 28 взаимоисключающие. Это означает, что может возникнуть две перегрузки, но отменить можно только одну из них.
29	Failed to synchronize IO Server time. Сбой синхронизации времени сервера ввода-вывода.	Ошибка		

*(продолжение следует)*

*(продолжение)*

№	Сообщение об ошибке	Тип	Описание	Комментарии
30	RT Server blocked for <nn> msec. (Сервер RT заблокирован на несколько секунд)	Критическая	Сервер (как потребитель данных) слишком занят, и поэтому драйвер ввода/вывода получает данные, минуя его. Это может означать серьезную угрозу потери данных.	
31	RT Server running again (Сервер возобновил работу).	Информационная	Блокировка сервера RT (ошибка 30) снята.	
32	Packet timestamp: (временная метка пакета) 10:24:32	Отладка	Показывает временную метку заголовка пакета.	Только в случае, если ключ реестра EL_SHOWTAGKEY > 0
33	Max string size reached: (Достигнут максимальный размер строки)	Предупреждение	Длина полученной строки превышает 255 символов.	
34	Max queue size reached: (Достигнут максимальный размер очереди) <nnn>, где <nnn> — число значений данных.	Информационная	Размер очереди превысил указанный в ключе реестра: MaxQueueSize [в Мб] (по умолчанию равен 15; минимальный — 1)	Очередь может быть одной из следующих: аналоговая, действительная, дискретная или текстовая.
1863	Topic connected. (Раздел подключен.) Отключение автоматической перегрузки	Информационная	Перегрузка SuiteLink будет отключена при подключении раздела SuiteLink.	

## П Р И Л О Ж Е Н И Е А

# Настройки реестра IndustrialSQL Server

Настройки системы IndustrialSQL Server хранятся в реестре Windows NT. В этом приложении описаны настройки реестра для IndustrialSQL Server и их стандартные значения.

## Содержание

- Редактирование реестра
- Ключи IndustrialSQL Server
- Ключи Microsoft SQL Server
- Ключи клиента IndustrialSQL
- Ключи FactorySuite
- Системные настройки

## Редактирование реестра

Информация реестра организована в иерархические группы, именуемые ключами, которые отображаются в виде пиктограмм в окне дерева. Каждый ключ может содержать одно или несколько значений.

Все параметры настройки IndustrialSQL Server находятся в ключах, расположенных под веткой реестра **HKEY\_LOCAL\_MACHINE**. Настройки клиентов IndustrialSQL Server находятся в ключах ветки **HKEY\_CURRENT\_USER**. В следующих описаниях для простоты мы опускаем название ветки реестра.

Для просмотра или изменения значений IndustrialSQL Server в реестре используйте редактор реестра Microsoft (REGEDT32.EXE для Windows NT; REGEDIT.EXE для Windows 95 или Windows 98).

Инструкции по работе с редактором реестра можно найти в документации Microsoft.

---

**Внимание!** Изменения в реестре Windows NT должны выполнять только опытные пользователи. Неправильное обращение с реестром может привести к серьезным проблемам конфигурации системы.

---

## Ключи IndustrialSQL Server

Ключ IndustrialSQL Server находится под веткой HKEY\_LOCAL\_MACHINE. Настройки IndustrialSQL Server хранятся в следующем ключе:

HKEY\_LOCAL\_MACHINE\SOFTWARE\Wonderware\InSQL

Ключ InSQL содержит следующие значения:

Значение ключа	Описание
NodeName:REG_SZ: <i>Имя системы</i>	(Только для внутреннего использования.)
NodeNumber:REG_DWORD: <i>Номер узла</i>	Номер, служащий для различения узлов хранения в многоузловой кластерной конфигурации. В конфигурации с одним узлом всегда равен 1.
UnitNumber:REG_DWORD: <i>Номер элемента</i>	(Только для внутреннего использования.)

## Пути хранения данных

Для всех узлов хранения в системе пути каталогов содержатся в следующем ключе:

\SOFTWARE\Wonderware\InSQL\Datapaths

---

**Примечание.** Все значения ключа Datapaths автоматически управляются службой Control Service на основе информации, полученной из InSQL Configure. **Не меняйте эти значения в реестре!**

---

Указанные пути также используются для выборки данных в одноузловой конфигурации. Ключ Datapaths содержит следующие значения:

Значение ключа	Описание
AltData:REG_SZ: <i>Путь альтернативного хранения</i>	Каталог, служащий в качестве альтернативного для циклической области хранения. При переполнении циклической области данные записываются в этот каталог.
AltDataMaxMBSize:REG_DWORD:0	Максимальный размер данных в Мб, который может вместить альтернативная область. При значении 0 используется все свободное дисковое пространство.
AltDataMinMBThreshold:REG_DWORD:125	Минимальное пространство логического диска в Мб, требуемое для альтернативной области хранения.
BufferDisk:REG_SZ: <i>Путь буферного хранения</i>	Каталог, в который переносятся ранее зарезервированные данные. Используется только системой выборки. Система хранения никогда не записывает в него.
DynData:REG_SZ: <i>Путь циклического хранения</i>	Каталог хранения архивных данных в реальном времени. Когда пространство в этом каталоге переходит за минимальный порог, система начнет перезаписывать самые старые данные, если только не определена альтернативная область - тогда старый блок данных будет перемещен в альтернативную область. (Если альтернативная область заполнена, самые старые блоки удаляются из нее по мере надобности.)
DynDataMaxMBSize:REG_DWORD:0	Максимальный размер данных в Мб, который может вместить циклическая область. При значении 0 используется все свободное дисковое пространство.
DynDataMinMBThreshold:REG_DWORD:125	Минимальное пространство логического диска в Мб, требуемое для циклической области хранения.

WormDisk:REG\_SZ:Путь постоянного хранения (архивирование) Каталог, в котором могут постоянно храниться архивные блоки. Это место, куда копирует данные хранимая процедура `xp_DiskCopy`.

## Программы

Имена программ, используемых для запуска системы хранения, и стандартные настройки регистрации ошибок хранятся под ключом:

`\SOFTWARE\Wonderware\InSQL\Programs`

## Регистрация ошибок

Стандартные параметры регистрации ошибок содержатся непосредственно в ключе Programs:

Значение ключа	Описание
EL_DEBUG:REG_DWORD:0	Сообщения только для отладки.
EL_DISPLAY:REG_DWORD:0	Информационные сообщения для конечного пользователя.
EL_INFO:REG_DWORD:3	Информационные сообщения для конечного пользователя.
EL_WARNING:REG_DWORD:7	Предупреждения для конечного пользователя.

Каждый тип сообщений имеет 3-битовое значение, в котором биты имеют следующий смысл:

Бит	Значение	Смысл
1	1	Сообщения выводятся на экран.
2	2	Сообщения записываются в текстовый файл журнала.
3	4	Сообщения записываются в журнал событий Windows NT.

Например, значение 7 включает все вышеуказанные режимы. Значение 3 задает вывод на экран и запись сообщений в файлы журнала.

Систему регистрации ошибок можно настроить на конкретную программу, добавив эти значения в соответствующий ключ программы. В этом случае значение в ключе программы будет перекрывать стандартное значение.

## Программные группы

Внутри ключа Programs имеется шесть подключей программных групп:

<b>Comms</b>	Коммуникационные программы драйвера и системы сбора данных.
<b>Event</b>	Программы системы управления событиями.
<b>InSQLIOS</b>	Сервер ввода-вывода реального времени

	IndustrialSQL .
<b>MsfmErf</b>	Обработчик различных данных (например, меток времени).
<b>Retrieval</b>	Сервер выборки базы данных.
<b>RTServer</b>	Уровень приема данных ввода-вывода .
<b>Storage</b>	Программы системы хранения данных.

Каждый из подключей программных групп имеет запись реестра (Numprogs), указывающую количество программ в группе. Каждый из этих подключей также содержит нумерованный подключ для каждой программы в группе. Например, первый драйвер описан в ключе:

```
HKEY_LOCAL_MACHINE
  \SOFTWARE
    \Wonderware
      \InSQL
        \Programs
          \Comms
            \1
```

Внутри каждого подключа программ имеется следующий минимальный набор значений. Эти значения описывают конфигурацию запуска каждой программы.

Значение ключа	Описание
Args:REG_SZ: <i>аргументы</i>	Аргументы командной строки, передаваемые программе при ее запуске.
Hidden:REG_DWORD: <i>скрытый</i>	1 = Запускается в скрытом режиме. 0 = Не запускается в скрытом режиме.
Maximised:REG_DWORD: <i>развернутый</i>	1 = Запускается в развернутом режиме. 0 = Не запускается в развернутом режиме.
Minimised:REG_DWORD: <i>свернутый</i>	1 = Запускается в свернутом режиме. 0 = Не запускается в свернутом режиме.
Progname:REG_SZ: <i>имя программы</i>	Полное имя программы.
Title:REG_SZ: <i>Заголовок</i>	Заголовок, отображаемый в главном окне программы.

Для некоторых программ, особенно для драйверов, могут требоваться дополнительные значения конфигурации. Например, для программ группы Comms дополнительно требуются следующие значения:

Значение ключа	Описание
ConnectTimeOut:REG_DWORD: <i>60</i>	Время в секундах, в течение которого драйвер ввода-вывода ждет подтверждения соединения от сервера ввода-вывода.
DeviceName:REG_SZ:	Имя устройства.

DeviceType:REG_DWORD:	Уникальный номер типа устройства. Берется из атрибута <i>IOServerTypeKey</i> в таблице <i>IOServer</i> .
GatewayName:REG_SZ:	Только для внутреннего использования.
MaxQueueSize:REG_DWORD:15	Максимальный размер в Мб очереди драйвера ввода-вывода. Минимально 1 Мб.
SourceId:REG_DWORD:	Уникальный номер драйвера. Берется из атрибута <i>IOServerKey</i> таблицы <i>IOServer</i> .
TimeSyncInterval:REG_DWORD: 60	Интервал в минутах для отправки на сервер ввода-вывода сообщения о синхронизации времени. По умолчанию - 1 час. Значение <i>TimeSyncInterval</i> равное 0 (нуль) отключает синхронизацию времени, т.е. сообщения не посылаются.
WriteTimeout:REG_DWORD: 5000	Время в мс, в течение которого драйвер воода-вывода будет ждать прежде, чем решит, что соединение с сервером ввода-вывода разорвано. Минимум - 1000 мс.

Программам, работающим как службы Windows NT, могут дополнительно требоваться следующие значения:

<b>Значение ключа</b>	<b>Описание</b>
Service:REG_DWORD:	Определяет, работает ли программа как служба. 1 = Работает как служба. 0 = Работает как консольное приложение.
ServiceArgs:REG_SZ:	Аргументы командной строки, передаваемые программе при ее запуске в качестве службы.
ServiceName:REG_SZ:	Имя службы.
ServiceLogon:REG_DWORD:	Учетная запись для запуска службы.

## Выборка данных

Параметры, относящиеся к базы данных *Runtime* и используемые для запуска IndustrialSQL Server и подсистемы выборки, находятся под следующим ключом:

\SOFTWARE\Wonderware\InSQL\Retrieval

Ключ Retrieval содержит следующие значения:

Значение ключа	Описание
DBName:REG_SZ: <i>Runtime</i>	Имя базы данных <i>Runtime</i> .
DefaultEndDate:REG_SZ: <i>yyyy-mm-dd hh:mm:ss</i>	Стандартная дата окончания выборки. Пока не используется.
DefaultPdsRowCount:REG_DWORD: <i>100</i>	Стандартное число строк, выбираемых системой IndustrialSQL Server. Во время выборки это значение может быть изменено с помощью расширения счетчика строк временного домена Industrial-SQL.
DefaultStartDate:REG_SZ: <i>yyyy-mm-dd hh:mm:ss</i>	Стандартная дата начала выборки. Пока не используется.
SQLServer:REG_SZ: <i>local</i>	Профиль клиента SQL, используемый системой IndustrialSQL Server для соединения с Microsoft SQL Server.
TimeOffset:REG_DWORD: <i>1</i>	Временное смещение, используемое системой IndustrialSQL Server, когда запрашиваются "живые" данные (в секундах).
TimeOffsetMult:REG_DWORD: <i>1</i>	Множитель, применяемый к временному смещению.

## Установка

В процессе установки IndustrialSQL Server значения параметров установки хранятся в следующем ключе:

\SOFTWARE\Wonderware\InSQL\Setup

Для полной установки ключ Setup содержит следующие значения:

Значение ключа	Описание
DataDir:REG_SZ:C:\InSQL\DATA	Путь к каталогу архивных блоков.
HoldingData:REG_SZ:15	Пространство (Мб), выделяемое для базы данных <i>Holding</i> .
HoldingLog:REG_SZ:10	Пространство (Мб), выделяемое для журналов транзакций в базе данных <i>Holding</i> .
InSQLDBPath:REG_SZ:C:\MSSQL7\DATA	Путь к каталогу, где хранятся файлы баз данных <i>Runtime</i> и <i>Holding</i> .
InstFolder:REG_SZ:FactorySuite\IndustrialSQL Server	Имя программной группы IndustrialSQL Server.
MainDir:REG_SZ: C: :\Program Files \InSQL	Корневой каталог файлов IndustrialSQL Server.
OdsPath:REG_SZ:нуть	Управляет работой Microsoft SQL Server с учетом некоторых требований лицензии Wonderware, если Microsoft SQL Server установлен с компакт-диска FactorySuite.
RemoteSQL:REG_SZ:0	0 = Не использовать удаленный MS SQL Server. 1 = Использовать удаленный MS SQL Server.
RunTimeData:REG_SZ:20	Пространство (Мб), выделяемое для данных в базе <i>Runtime</i> .
RunTimeLog:REG_SZ:10	Пространство (Мб), выделяемое для журналов транзакций в базе данных <i>Runtime</i> .
ServerName:REG_SZ:local	Указывает, работает ли Microsoft SQL Server локально или удаленно. В локальном режиме имя сервера будет <i>local</i> . В удаленном режиме именем сервера будет имя компьютера с удаленной системой Microsoft SQL Server.

## Хранение данных

Параметры, относящиеся к базы данных *Runtime* и используемые для запуска IndustrialSQL Server и подсистемы хранения, находятся под следующим ключом:

\SOFTWARE\Wonderware\InSQL\Storage

Ключ Storage содержит следующие значения:

Значение ключа	Описание
1ByteSnapshotSize:REG_DWORD:	Размер снимка, по которому система определяет необходимость запуска следующего снимка. Размер зависит от количества тэгов и частоты их сохранения.  См. главу 4, "Управление системой хранения".
2ByteSnapshotSize:REG_DWORD:	(см. выше)
4ByteSnapshotSize:REG_DWORD:	(см. выше)
8ByteSnapshotSize:REG_DWORD:	(см. выше)
AIAutoResize:REG_DWORD:1	Управляет автоматическим изменением размера активного образа.  1 = Включено. 0 = Отключено.
AIResizeSecInterval:REG_DWORD:300	Скорость в секундах, с которой система пересчитывает число выборок, используемых для каждого тэга в активном образе.
AnalogQualitySnapshot:REG_DWORD:	Используется для дельта-хранения и хранения данных качества.
Compression:REG_DWORD:1	Коэффициент сжатия сохраняемых данных.
DiscreteQualitySnapshot:REG_DWORD:	Используется для дельта-хранения и хранения данных качества.
NumDataSources:REG_DWORD:2	Количество источников данных.
QualitySize:REG_DWORD:2	Размер значения качества.
SecondsPerBlock:REG_DWORD: 86400	Размер (сек.) архивного блока. Стандартный размер - 24 часа. Размер архивного блока должен быть всегда больше максимальной скорости сканирования.
VarSize:REG_DWORD:4	Размер хранения тэгов (2 или 4 байта).

## Скорость сканирования

Параметры скорости сканирования хранятся под следующим ключом:

\SOFTWARE\Wonderware\InSQL\Storage\Scanrates

Ключи и значения реестра не заполняются, пока не запущена система хранения. Ключ Scanrates содержит следующие значения:

Значение ключа	Описание
Comment:REG_SZ: <i>1 second scanrate</i>	Комментарий, указывающий скорость сканирования в секундах.
Rate:REG_DWORD: <i>3e8</i>	Скорость сканирования в миллисекундах.
Suffix:REG_SZ: <i>1</i>	Идентификатор файла.

## Ключи Microsoft SQL Server

Одна единственная запись о системе IndustrialSQL Server хранится в ветке реестра HKEY\_LOCAL\_MACHINE. Эта запись находится в ключе:

\SOFTWARE\Microsoft\MSSQLServer\InSQL

Это ключ содержит следующее значение:

Значение ключа	Описание
ListenOn:REG_MULTI_SZ: <i>SSMNP70\.\pipe \\wssql</i>	Канал Named pipe, через который IndustrialSQL принимает запросы клиентов.

## Ключи клиента IndustrialSQL

Ключ реестра для установки клиентов IndustrialSQL находится под веткой HKEY\_LOCAL\_MACHINE. Параметры установки клиента IndustrialSQL хранятся под следующим ключом:

HKEY\_LOCAL\_MACHINE\SOFTWARE\Wonderware\InSQLClient\Setup

Ключ InSQLClient\Setup содержит следующие значения:

Значение ключа	Описание
InstFolder:REG_SZ:Путь	Имя программной группы IndustrialSQL Server.
MainDir:REG_SZ:Путь	Корневой каталог файлов клиента IndustrialSQL.

Значения реестра для каждого клиентского приложения хранятся под веткой HKEY\_CURRENT\_USER. Параметры клиентов IndustrialSQL находятся в следующем ключе:

HKEY\_CURRENT\_USER\SOFTWARE\Wonderware\InSQL

Для каждого клиентского приложения в соответствующих папках хранятся следующие параметры:

- Параметры соединения с базой данных.
- Параметры конфигурации программ.

Параметры InSQL Control и соответствующей службы хранятся в следующих ключах:

HKEY\_CURRENT\_USER\SOFTWARE\Wonderware\InSQL\InSQL Service Control

HKEY\_CURRENT\_USER\SOFTWARE\Wonderware\InSQL\InSQLCtrl

## Ключи FactorySuite

Для каждого компонента FactorySuite имеется запись реестра под веткой HKEY\_LOCAL\_MACHINE:

\SOFTWARE\Wonderware\FS2000\Products

Ключ Products содержит следующие значения IndustrialSQL:

Значение ключа	Описание
InSQLClient:REG_DWORD:1	1 = Клиенты IndustrialSQL установлены. 0 = не установлены.
InSQLServer:REG_DWORD:1	1 = IndustrialSQL Server установлен. 0 = не установлен.

## Системные настройки

### Регистрация ошибок

Параметры регистрации ошибок в журнале событий Windows NT хранятся под следующим ключом:

\SYSTEM\CurrentControlSet\Services\EventLog\Application\

Имеется четыре ключа для IndustrialSQL Server: InSQL, InSQL Control, InSQL DbServer и InSQL IODriver. Эти ключи содержат следующие значения:

Значение ключа	Описание
EventMessageFile:REG_EXPAND_SZ: <i>путь</i>	Путь файла .DLL, обрабатывающего сообщения.
TypesSupported:REG_SZ:7	Типы поддерживаемых событий. Значение должно быть 7.

### Службы

Эти значения управляются диспетчером служб NT. Никакие из них не следует изменять. Параметры, необходимые для работы процессов IndustrialSQL Server в качестве служб хранятся под следующим ключом:

\SYSTEM\CurrentControlSet\Services

#### Службы InSQL

В ключе Services имеется пять подключей IndustrialSQL:

<b>InSQL Control</b>	Программа IndustrialSQL Control, используемая для запуска и остановки системы.
<b>InSQL DbServer</b>	Сервер выборки данных. Этот процесс может запускаться и останавливаться независимо от остальной системы. Когда эта служба остановлена, клиенты не могут подключаться к IndustrialSQL Server. Однако процесс сохранения данных продолжается.
<b>InSQL Event System</b>	На основе конфигурации системы распознает события и выполняет ответные действия.
<b>InSQL IODriver</b>	Считывает базу данных IndustrialSQL Server и выполняет инициализацию активного образа. Обеспечивает передачу данных из драйверов устройств (серверов ввода-вывода) в систему сбора.
<b>InSQLIOS IOServer</b>	Интерфейс клиентов для доступа к данным "активного образа" IndustrialSQL Server через протокол DDE или SuiteLink. Сервер ввода-вывода InSQL может обновлять элементы текущими значениями указанных тем, играя роль сервера "реального времени".

В программных подключаемых имеет следующий набор значений. Не каждая служба пользуется всеми ключами. Эти ключи не могут изменяться пользователем.

Значение ключа	Описание
DependOnGroup:REG_MULTI:	Группа служб, от которых зависит запуск программы.
DependOnService:REG_MULTI:	Служба, от которой зависит запуск программы. Если служба не запущена, ее попытается запустить система NT.
DisplayName:REG_SZ:	Имена служб, отображаемые в Панели управления.
ErrorControl:REG_DWORD:	Действия NT при сбое запуска службы.
ImagePath:REG_EXPAND_SZ:	Путь к исполняемому файлу службы.
ObjectName:REG_SZ:	Учетная запись, с которой запускается служба.
Start:REG_DWORD:	Указывает, запускается ли служба вручную, автоматически или она отключена.
Type:REG_DWORD:	Указывает, относится ли служба к исполняемому типу файлов.
Group:REG_SZ:	Группа, к которой относится служба.

Каждый из этих подключей программных групп содержит две папки: **Enum** и **Security**. Значения в папке Enum следующие:

Значение ключа	Описание
0:REG_SZ:	Используется устройствами plug-n-play.
Count:REG_DWORD:	Используется устройствами plug-n-play.
NextInstance:REG_DWORD:	Используется устройствами plug-n-play.

Значения в папке Security следующие:

Значение ключа	Описание
Security:REG_BINARY:	Относится к имени учетной записи. Не расшифровывается.

## П Р И Л О Ж Е Н И Е В

## Толковый словарь

<i>binary</i> тип данных	<i>Бинарные</i> данные есть тип данных, используемый для бинарных строк или битовых выборок. Бинарные строки вводятся шестнадцатиричным кодом. Есть два типа бинарных данных: <code>binary[(n)]</code> и <code>varbinary[(n)]</code> . См. также <code>binary[(n)]</code> и <code>varbinary[(n)]</code> .
<i>binary[(n)]</i> тип данных	Тип данных <i>binary(n)</i> может содержать до 255 байт бинарных данных с фиксированной длиной. Максимальная длина указана через <i>n</i> , где значение <i>n</i> варьируется от 1 до 255. Размер хранения для <i>бинарных</i> данных есть значение <i>n</i> . <i>Бинарные</i> данные — это тип системных данных. См. также <code>varbinary[(n)]</code> .
<i>bit</i> тип данных	<i>Битовые</i> данные могут содержать 1 или 0. Любое целое значение, не равное 0, автоматически понимается как 1. <i>Битовые</i> данные используются для значений да/нет и истина/ложь. Размер хранения для <i>битовых</i> данных 1 байт; вместе с тем, в одном байте могут быть сохранены до восьми атрибутов. См. также дискретное значение.
BLOB	См. крупный бинарный объект
<i>char</i> тип данных	Символьные данные есть тип данных, который может содержать до 255 символов. Символьные данные имеют постоянную длину хранения ( <i>n</i> ).
COM	См. модель составного объекта.
CPS	См. символов в секунду.
<i>datetime</i> тип данных	Данные <i>datetime</i> содержат дату и время суток. Размер хранения для данных <i>datetime</i> 8 байт, которые разделены на 4-байтовые целые: одно 4-байтовое целое для количества дней до или после 1 января 1900 года, и одно 4-байтовое целое для количества миллисекунд после полуночи. Даты до 1 января 1900 года сохраняются как отрицательные значения. <i>datetime</i> есть системный тип данных.
DBO	См. владелец базы данных.
DCOM	См. Распределенная модель составного объекта
FK	См. внешний ключ.

---

<i>float</i> тип данных	Данные типа <i>float</i> могут содержать положительные или отрицательные числа с плавающей точкой с точностью до 15 знаков. Значения могут быть от $1.7E - 308$ до $1.7E 308$ . Размер хранения — 8 байт. <i>float</i> есть системный тип данных.
HTTP	См. <i>Протокол передачи гипертекста</i> .
IDENTITY столбец	Столбец идентификации содержит созданное системой значение, используемое для уникальной идентификации каждой строки таблицы. Если данные помещаются в таблицу, где определен столбец идентификации, сервер SQL автоматически создает значение. Это значение основано на последнем значении идентификации плюс приращение заранее определенного значения идентификации.
<i>image</i> тип данных	Данные типа <i>image</i> могут содержать от 0 до 2 147 483 647 байт бинарных данных различной величины. Данные <i>image</i> не могут использоваться для переменных или параметров в хранимых процедурах. <i>image</i> — это системный тип данных.
<i>int</i> тип данных	Данные типа <i>int</i> есть целые данные, которые могут содержать целые числа от $2^{31} - 1$ (то есть, 2 147 483 647) до $-2^{31}$ (то есть, -2 147 483 648), включительно. Размер хранения — 4 байта. <i>int</i> есть системный тип данных.
IPX/SPX	IPX/SPX есть протокол передачи данных, используемый в сетях Novell®.
Named pipes	Механизм межпроцессового обмена (IPC), используемый для передачи данных между отдельными процессами, работающими обычно на разных компьютерах. В протоколе named pipes канал передачи (pipe) стабилизируется усилиями обоих процессов. См. также межпроцессовый обмен.
NULL	NULL означает, что запись в столбце не имеет соответствующего ей значения. NULL <i>не</i> означает, что числовое значение равно 0, не означает пустую строку. NULL есть отсутствие значения. Если для столбца не разрешено использование NULL, значение должно быть указано.
ODBC	См. Соединение открытых баз данных Microsoft.
ODS	См. <i>Службы открытых данных</i> .

---

OLE DB	Object Linking and Embedding for Databases (OLE DB) есть интерфейс программирования приложений (API), позволяющий приложениям-клиентам на COM-основе получать доступ к данным, не хранящимся физически в сервере Microsoft SQL, с которым они соединяются.
OLE DB	См. <i>Object Linking and Embedding for Databases</i> .
PK	См. первичный ключ.
<i>real</i> тип данных	Данные типа <i>real</i> ( <i>действительные</i> ) могут содержать положительные или отрицательные числа с плавающей точкой, с точностью до 7 знаков. Значения данных <i>float</i> могут быть от 3.4E - 38 до 3.4E 38. Размер хранения для данных типа <i>float</i> — 8 байт. <i>float</i> есть системный тип данных.
RI	См. целостность связей.
<i>smalldatetime</i> тип данных	Данные типа <i>smalldatetime</i> содержат дату и время суток с меньшей точностью, чем <i>datetime</i> . Размер хранения для данных <i>smalldatetime</i> — 4 байта. Данные <i>smalldatetime</i> состоят из одного малого целого числа для количества дней после 1 января 1900 и одного малого целого числа для количества минут после полуночи. Дата варьируется от 1 января 1900 до 6 июня 2079 с точностью до минуты. <i>smalldatetime</i> есть системный тип данных.
<i>smallint</i> тип данных	Данные типа <i>smallint</i> могут содержать целые числа между 215 - 1 (32 767) и -215 (-32 768), включительно. Размер хранения для данных <i>smallint</i> — 2 байта. <i>smallint</i> есть системный тип данных.
SQL	См. Язык структурированных запросов
SuiteLink	Сетевой протокол SuiteLink предназначен для высокопроизводительных промышленных приложений. SuiteLink располагает функцией Value Time Quality (VTQ) и выставляет штамп времени и качества на каждое значение данных, передаваемое совместимым с VTQ клиентам. SuiteLink работает на базе протокола TCP/IP.
TCP/IP	TCP/IP (Transmission Control Protocol/Internet Protocol) есть набор сетевых протоколов, обеспечивающих передачу пакетов данных через разнородные сети. TCP/IP может передавать пакетную информацию между различными видами оборудования и операционных систем.

<i>text</i> тип данных	Данные типа <i>text</i> могут содержать до 231 - (2 147 483 647) символов. Размер хранения для данных типа <i>text</i> может быть до 2 Кб, если они сохраняются не как BLOB. <i>text</i> является системным типом данных. См. также крупный бинарный объект.
<i>tinyint</i> тип данных	Данные типа <i>tinyint</i> могут содержать любое число от 0 до 255, включительно. Размер хранения для данных типа <i>tinyint</i> — 1 байт. <i>tinyint</i> является системным типом данных.
<i>varbinary</i> [( <i>n</i> )] тип данных	Данные типа <i>varbinary</i> ( <i>n</i> ) могут содержать от 0 до 255 включительно бинарных данных различной длины, где <i>n</i> указывает максимальный размер столбца. Размер хранения для данных <i>varbinary</i> есть размер значений данных для этого столбца, а не значение <i>n</i> . Данные <i>varbinary</i> являются системным типом данных.
<i>varchar</i> [( <i>n</i> )] тип данных	Данные типа <i>varchar</i> ( <i>n</i> ) могут содержать от 1 до 255 включительно символов данных различной длины. Столбец <i>varchar</i> может содержать 0 символов. Размер хранения для данных типа <i>varchar</i> есть размер значений данных, заполняющих столбец. Данные типа <i>Varchar</i> должны быть заключены в одинарные кавычки. Данные типа <i>varbinary</i> являются системным типом данных.
WORM	WORM есть оптический диск, запись на который возможна только однократно. WORM расшифровывается как "Write Once, Read Many" (записать однократно, считать многократно).
Агрегация	Функции агрегации есть функции SQL, которые проводят числовые подсчеты со столбцами в наборе данных. Функции агрегации: SUM, AVG, MIN, MAX и COUNT.
Активный образ	Активный образ есть область памяти, в которой временно содержится определенное количество значений полученных данных до их сохранения на диск.
Альтернативная область хранения	Альтернативная область хранения есть каталог на компьютере, используемый для хранения файлов, перемещенных их циркулирующей области хранения.
Аналоговое значение	Аналоговое значение есть переменная, измеряющая непрерывную физическую величину. Например, температура котла будет измеряться как аналоговое значение.

---

Аннотация	Аннотация есть пользовательский комментарий о тэге за определенный момент времени.
Архивировать	Архивировать —перемещать данные на устройства длительного хранения, такие как DAT-кассеты. Архивирование данных предотвращает их потерю в случае удаления или уничтожения исходных данных. <i>См. также резервировать.</i>
Архивные таблицы	В IndustrialSQL Server архивные таблицы представляют полученные производственные данные в архивном формате, где находится одна строка для каждого сохраненного значения тэга. <i>См. также живые таблицы, широкие таблицы.</i>
Архивный блок	Архивным блоком называется группа файлов хранения данных, которые записываются в отдельный каталог, идентифицируемый штампом даты и буквенным суффиксом. IndustrialSQL Server сохраняет полученные данные на диск в блоках. Архивные блоки создаются, когда система запускается или после истечения заранее определенного и указанного в реестре Windows NT периода времени.
Атрибут	В базе данных атрибуты представляют характеристики или свойства ячейки, такие как тэг, и обычно относятся к заголовкам столбцов таблицы.
База данных	База данных — это системное хранилище данных общих типов, организованных с помощью уникальных идентификаторов и таблиц. Базы данных хранятся в файлах, называемых устройствами баз данных.
База данных <i>runtime</i>	В IndustrialSQL Server база данных <i>runtime</i> содержит таблицы, хранящие все данные конфигурации, архивов и текущих процессов. <i>См. также база данных Holding.</i>
База данных <i>Holding</i>	В IndustrialSQL Server база данных <i>Holding</i> содержит таблицы, используемые для временного хранения информации, импортированной с узла InTouch перед помещением ее в базу данных <i>runtime</i> . <i>См. также база данных runtime.</i>
Байт	Байт есть единица информации, состоящая из 8 бит. С точки зрения хранимых данных, байт равен одному символу, как например буква или цифра.
Блок	<i>См. архивный блок.</i>

Броузер	Броузер есть графическое представление иерархий данных. Броузер используется для показа каталогов на диске, папок, файлов и т.п. Клиенты IndustrialSQL Server используют броузер для показа серверов, тэгов и групп тэгов в системных, личных и общих именных пространствах. <i>См. также</i> именные пространства.
Булево выражение	Булево выражение выдает значение истины или лжи. Булево выражение содержит логические операторы, такие как И, ИЛИ и НЕ. <i>См. также</i> логические операторы.
Буферная область хранения	Буферная область хранения есть каталог на компьютере, используемый для временного хранения файлов, например, для получения данных из архива.
Ведомый край	Если в расширении определения края указан параметр "Ведомый", это означает, что единственными строками в результатах запроса будут те, которые не совпали с критериями (выдана ложь) после строки, которая с этими критериями совпала (выдана истина). <i>См. также</i> ведущий край, определение края.
Ведущий край	Ведущий край — это когда запрос выдает только те строки, которые первыми удовлетворили критериям (выдана истина) после строки, этим критериям не удовлетворившей (выдана ложь). <i>См. также</i> определение края, ведомый край.
Владелец базы данных (DBO)	Владелец базы данных есть пользователь, создающий базу данных. Владельцы баз данных имеют право полного доступа к созданным ими базам данных, включая право на предоставление доступа к этим базам данных другим пользователям. У каждой базы данных может быть только один владелец. Владение базой данных может передаваться между пользователями и владелец базы данных может иметь несколько идентификационных псевдонимов.
Владелец объекта	<i>См. владелец объекта базы данных.</i>
Владелец объекта базы данных	Владелец объекта базы данных — это пользователь, создающий объект базы данных, такой как таблица, указатель, представление, правило и т.п. Владельцы объектов баз данных имеют полный доступ ко всем созданным ими объектам, включая право на предоставление доступа к этому объекту другим пользователям. Право на владение объектом базы данных не передается.

---

Внутренний тэг	Внутренние тэги — это тип тэгов, существующих внутри приложения InTouch. Они используются для создания постоянных системных значений и моделей. Также они используются при создании подсчитываемых переменных, к которым обращаются другие программы Windows. Существует 4 типа внутренних тэгов: внутренний дискретный, внутренний целый, внутренний действительный и внутренний текстовый.
Временной интервал	Для системы управления событиями временным интервалом называется скорость, с которой детекторы событий проверяют, произошло ли событие. Этот интервал времени также известен как скорость сканирования.
Граница	Границей называется определяемое пользователем минимальное или максимальное значение для отрезка значений.
Группа	См. группа пользователей.
Группа пользователей	Группой пользователей называется группа пользователей базы данных, имеющих право на определенные действия с базой данных. Группы пользователей являются частью системы безопасности базы данных. Любой пользователь базы данных, добавляемый в группу пользователей, получает права, соответствующие этой группе.
Дамп	См. резервирование.
Дамп на ленту	Дамп на ленту есть копирование структур базы и хранимых данных на ленточное устройство.
Данные	Кодированное представление информации для использования в компьютере. У данных есть атрибуты, такие как тип и длина.
Действие	Действие над событием есть действие, совершаемое в момент распознавания детектором возникновения события. Действия над событиями необязательны; бывают случаи, когда нужно только зафиксировать возникновение события. См. также детектор, тэг события.
Действительное число	Действительное число есть число с плавающей точкой, представленное цифрами с фиксированной основой, как например в десятичной системе. Действительное число может быть составлено из конечного или бесконечного количества цифр.

---

Дельта-выборка	Дельта-выборка, или выборка методом исключения, есть получение изменения значения тэга в заданный период времени. То есть, дублируемые значения не выдаются. <i>См. также</i> циклическая выборка.
Дельта-хранение	Дельта-хранение есть сохранение данных, основанное на изменении значения. Дельта-хранение позволяет сохранять тэг только при изменении значения. Дельта-хранение также называется "хранение методом исключения". Этот метод используется для сохранения дискретных или аналоговых значений, скорость изменения которых неустойчива (иногда быстра, иногда медленна) и когда требуется дискретность менее 1 секунды. <i>См. также</i> циклическое хранение.
Детектор	Детектор событий есть механизм определения, когда были удовлетворены критерии событий для тэга события. Детекторы событий срабатывают в соответствии с интервалом времени.
Динамический обмен данными (DDE)	DDE есть процесс передачи данных между приложениями без участия пользователя и без просмотра. В среде Windows DDE реализован в виде набора сообщений, рекомендуемых процедур (протоколов) для обработки этих сообщений и некоторых новых типов данных. Следуя этим протоколам, приложения, написанные независимо друг от друга, могут передавать данные друг другу без участия пользователя. Например, InTouch и Excel. <i>См. также</i> тема, элемент, общий ресурс DDE, сервер ввода-вывода, клиент DDE, SuiteLink.
Дискретное	Дискретное значение есть переменная, имеющая только два состояния: '1' (Истина, Включено) или '0' (Ложь, Выключено).
Дискретность	Дискретность есть интервал в миллисекундах для получения данных из любой аналоговой или дискретной архивной таблицы IndustrialSQL Server. Расширение временного домена дискретности IndustrialSQL Server не поддерживается обычными функциями сервера SQL. Количество выдаваемых строк зависит от периода времени запроса и дискретности (количество строк = период времени / дискретность). Дискретность применяется только при циклической выборке.
Допуск к объекту	Допуск к объекту определяет, какие предложения могут быть использованы с объектом базы данных. Допуск к объекту определяется владельцем этого объекта базы данных. <i>См. также</i> объект, владелец объекта базы данных.

Драйвер ODBC	Драйвер ODBC позволяет ODBC-совместимому приложению обращаться к источнику данных. Драйвер ODBC является библиотекой DLL. <i>См. также</i> Соединение открытых баз данных Microsoft, Библиотека динамического соединения.
Единица измерения	Единица измерения для значений тэга. Например, RPM, миллисекунды, градусы.
Живой	"Живой" — термин, описывающий данные, отражающие самое последнее значение тэга.
Живые таблицы	В IndustrialSQL Server "живые" таблицы содержат текущие значения полученных производственных данных для аналоговых, дискретных или текстовых тэгов. <i>См. также архивные таблицы, широкие таблицы.</i>
Журнал	Таблица данных, разделенная на категории по простым типам информации, которая хранится в базе данных и использует штамп времени как первичный ключ.
Журнал транзакций	Журнал транзакций есть запись всех изменений базы данных в базу данных. <i>См. также</i> транзакция.
Задержка	Задержка — это количество времени между действительным происхождением события в системе и его распознаванием детектором событий.
Запись	<i>См.</i> строка.
Запрос	Запрос есть сценарий выражений SQL, подаваемый в базу данных клиентом, который выполняет поиск объектов в таблице базы данных.
Запрос к базе данных	<i>См.</i> запрос.
Идентификатор пользователя	<i>См.</i> учетная запись.
Именное пространство	Именное пространство есть имеющий имя набор объектов. Именное пространство — это логическая "область", содержащая иерархию объектов. Например, серверы, тэги, темы и т.п. IndustrialSQL Server определяет три вида именных пространств: системные, общие и личные. Иерархическое содержание именного пространства отображается в броузере или приложении-клиенте. В IndustrialSQL Server определение, заключенное в именное пространство, управляется внутренними хранимыми процедурами. <i>См. также</i> системное именное пространство, общее именное пространство, именное пространство, браузер, хранимые процедуры.

---

Имя базы данных	Имя базы данных используется для идентификации базы данных при подключении к ней клиента.
Имя источника данных (DSN)	По стандарту Связи открытых баз данных (ODBC) это имя соединения, при помощи которого клиент обращается к серверу. <i>См. также</i> Связь открытых баз данных.
Имя пользователя	Имя пользователя идентифицирует пользователя базы данных в целях безопасности. Имя пользователя добавляется к учетной записи для доступа конкретного пользователя к базе данных.
Имя сервера	Именем сервера называется имя, присвоенное IndustrialSQL Server. Имя сервера должно соответствовать правилам идентификации SQL.
Имя тэга	Имя тэга есть имя, присвоенное переменной в базе данных.
Индекс	Индексом называется набор указателей, которые дают более быстрый доступ к данным в строках таблицы, чем при сканировании таблицы. Понятие индекса таблицы близко к предметному указателю в конце этой книги; записи в предметном указателе делают поиск данных быстрее, чем при пролистывании книги от начала в поисках нужной информации. Индексы также могут усилить унификацию строк в таблице. Существует два типа индексов: сгруппированные и несгруппированные. <i>См. также</i> сгруппированный индекс, несгруппированный индекс.
Инициализация	Инициализация есть запуск IndustrialSQL Server и систем хранения.
Внешний ключ (FK)	Внешний ключ есть один или несколько столбцов, чьи значения совпадают с первичным ключом (PK) какой-либо другой таблицы. Одиночный первичный ключ может иметь внешние ключи более чем в одной таблице.
Интерфейс человека с машиной (HMI)	Интерфейс человека с машиной есть интерфейс программного обеспечения, позволяющий операторам производственных участков просматривать, манипулировать и сохранять производственные данные. HMI может быть запущен на PC или другом производственном терминале.
Каскадировать	При редактировании ячейки базы данных (например, при операциях UPDATE или DELETE) используется для автоматического проведения того же действия над всеми связанными строками и столбцами базы данных.

Качество	Качеством называется показатель точности, доступности и правильности полученных данных. Значения данных, сохраняемые в IndustrialSQL Server, имеют соответствующее значение качества.
Кластерный индекс	В кластерном индексе логический порядок значений символов аналогичен физическому порядку данных. То есть данные содержатся в порядке, предусмотренном кластерным индексом. В таблице может быть только один кластерный индекс, так как может быть только один физический порядок строк. Индексы используются системой для унификации строк или для улучшения процесса поиска данных в базе.
Клиент DDE	Клиент DDE есть приложение, получающее данные от сервера ввода-вывода через протокол Microsoft Dynamic Data Exchange. <i>См. также</i> Динамический обмен данными.
Контекст	Смысловое описание события или связи, которое может охватывать группу пределов, скорости изменения или отклонения. Примерами являются "Нормальная работа", "Холодное выключение", "Особое внимание".
Крупный бинарный объект (BLOB)	Крупный бинарный объект, или BLOB, есть сложное значение данных, такое как массив, файл .AVI или рисунок, которое не интерпретируется базой данных.
Кэш	Кэш есть специальная подсистема памяти, в которой сохраняются данные, к которым часто происходит обращение, и соответствующие им адреса. Кэш используется как буфер для хранения данных во время перемещения между жестким диском и ОЗУ. Обычно кэш-память быстрее чем ОЗУ.
Личное именованное пространство	Личное именованное пространство есть набор пользовательских производственных компонентов, таких как производственные участки, имена компьютеров и тэги, связанные с конкретным компьютером или процессом. Личное именованное пространство работает также, как и общее именованное пространство, за исключением того, что личное именованное пространство доступно не всем пользователям системы IndustrialSQL Server. Личное именованное пространство создается пользователем IndustrialSQL Server. Иерархия содержимого личного именованного пространства может быть просмотрена в дереве каталога, но будет видна только пользователю, создавшему ее. <i>См. также</i> именованное пространство, системное именованное пространство, общее именованное пространство, браузер.

---

Логические операторы	Логические операторы используются для подсчета или сравнения данных. Образцами логических операторов являются И (AND), ИЛИ (OR), и НЕ (NOT). Логические операторы И, ИЛИ и НЕ могут использоваться в выражениях WHERE для указания условий поиска. Оператор И означает, что оба условия выполнены. ИЛИ означает, что выполнено одно из условий. НЕ означает, что выполнены все условия, кроме тех, на которые указывает этот оператор. <i>См. также</i> булево выражение.
Логические таблицы	<i>См.</i> представление, таблица расширения.
Логическое представление	<i>См.</i> представление.
Межпроцессовая связь (IPC)	Межпроцессовая связь есть система, при которой один процесс или задание может обмениваться данными с другим. <i>См. также</i> named pipe
Мертвая зона	Мертвая зона есть величина увеличения или уменьшения значения, которая допускается до происхождения события в системе.
Место хранения	Место хранения есть каталог, в который сохраняются файлы архивных данных.
Миллисекунда	Одна тысячная доля секунды, сокращенно мс или мсек.
Множественный протокол	Сетевая библиотека множественных протоколов позволяет серверу принимать входящие соединения сети по протоколам named pipes, портам TCP/IP и слотам SPX. Если для получения имен через поддерживаемые протоколы используется локальная база данных RPC, то в строке соединения необязательно указывать порты и слоты. <i>См. также</i> межпроцессовый обмен, вызовы удаленных процедур, слот.
Модель составного объекта (COM)	COM есть набор служб, которые позволяют компонентам программного обеспечения совместно работать в сетевой среде.
Набор строк	Набор строк есть строки, каждая из которых содержит столбцы данных.
Необработанное значение	Необработанным называется значение элемента данных при его получении. Подсчеты и преобразования могут проводиться с необработанными значениями до использования этих значений системой IndustrialSQL Server.

Несгруппированный индекс	Несгруппированным называется индекс, в котором логический порядок номеров не совпадает с физическим порядком строк в таблице. Несгруппированный индекс есть группа логических ссылок на строки. В отличие от сгруппированного индекса, для одной таблицы может быть определено до 249 несгруппированных индексов. Обычно несгруппированные индексы работают медленнее сгруппированных. Индексы используются для быстрого поиска информации в таблицах. <i>См. также</i> индекс, сгруппированный индекс.
Общее именованное пространство	Общее именованное пространство есть созданный администратором набор производственных компонентов. Так же, как системное именованное пространство включает в себя информацию о системе IndustrialSQL Server, общее именованное пространство включает в себя информацию об участке, на котором запущен IndustrialSQL Server. Общее именованное пространство включает в себя такие объекты, как производственные участки, имена компьютеров и тэги, связанные с конкретным компьютером или процессом. Общее именованное пространство создается администратором IndustrialSQL Server. Иерархия содержимого общего именованного пространства видна всем пользователям в дереве каталога. <i>См. также</i> именованное пространство, системное именованное пространство, личное именованное пространство, браузер.
Общий ресурс DDE	Общий ресурс DDE есть конфигурация памяти компьютера, доступ к которой имеют приложения с другого компьютера. Общий ресурс DDE используется для взаимодействия между несколькими DDE-совместимыми приложениями. Операции DDE также называются "диалогами" DDE. <i>См. также</i> Динамический обмен данными, тема, элемент.
Объект	Объектом называется любой из компонентов, составляющих базу данных. Примерами объектов базы данных могут служить таблицы, представления, символы, стандарты, триггеры, индексы, хранимые процедуры. Также называется объектом базы данных.
Объект базы данных	<i>См.</i> объект.
Ограничение	Ограничение есть запрет на выдачу данных, который может быть вставлен в столбец или строку.
Определение края	Определение края для конкретных данных. Край есть абстрактная линия, где удовлетворение критериям изменялось с истины на ложь и наоборот. <i>См. также</i> ведущий край, ведомый край.

---

Первичный ключ (РК)	Первичным ключом называются один или несколько столбцов, которые делают строку таблицы уникальной. Первичный ключ используется при связке с внешними ключами в других таблицах. <i>См. также</i> Внешний ключ.
Отклонение	Отклонение есть процент изменения значения тэга по отношению к зафиксированному значению, называемому целью. У каждого аналогового тэга могут быть два возможных отклонения: высшее и низшее.
Пара сообщений	Состояния, связанные с состояниями ИСТИНА (Включено) или ЛОЖЬ (Выключено) дискретного значения. <i>См. также</i> дискретное значение.
Параметр	Параметр есть информационный элемент, имеющий значение. Параметры определяют, требуется ли записать значение или выдать его из базы данных.
Повторная инициализация	Повторная инициализация есть перезапуск одного или нескольких процессов в системе IndustrialSQL Server. При редактировании базы данных <i>Runtime</i> может понадобиться перезагрузка для того, чтобы изменения возымели эффект.
Поле	<i>См.</i> столбец.
Получатель	Для сложного тэга получателем будет функция или способ получения данных, хранимых в базе данных. Например, правильный поток данных. Функции получателя хранятся в библиотеке DLL в системе IndustrialSQL Server. <i>См. также</i> поставщик данных, сложный тэг, библиотека динамического соединения.
Пользовательские данные	Пользовательскими называются данные, созданные пользователем. Пользовательские данные существуют в дополнение к уже существующим системным типам данных. Данные, которые могут быть сохранены в столбце таблицы базы данных, определяются типом данных для этого столбца. Стандарты и правила применяются только к пользовательским данным, а не к системным типам.
Порядок сортировки	Порядок сортировки есть набор правил, которые указывают, как сервер SQL должен упорядочивать и представлять данные в качестве результата запроса базы данных. Порядок сортировки определяет, как сервер SQL будет обращаться со сравнениями элементов для хранения данных и операций выборки данных при помощи выражений GROUP BY, ORDER BY и DISTINCT. Сервер SQL также использует выделенный порядок сортировки для получения запросов при помощи выражений WHERE и DISTINCT.

---

Поставщик данных	Для сложного тэга поставщик данных есть функция поставки данных для сохранения в базе данных. Например, правильный поток данных. Функции поставщика данных хранятся в библиотеке DLL в системе IndustrialSQL Server. См. также получатель, сложный тэг, библиотека динамического соединения.
Поставщик данных OLE DB	"Виртуальный" сервер, позволяющий внутренней программе обращаться к хранилищу данных OLE DB.
Постоянная область хранения	Постоянная область хранения есть каталог на компьютере, используемый для хранения критических данных (например, работы реактора), которые нельзя перезаписывать.
Права доступа	Права доступа ограничивают пользователя базы данных в действиях, которые он может проводить с базой данных. Например, пользователь может иметь право на выборку данных из всех таблиц БД, но не иметь прав на запись данных.
Правило	Правило базы данных есть объект, привязанный к столбцу таблицы или пользовательским данным. Правила определяют, какие типы данных могут быть введены в таблицу. Например, правило может указывать, что количество часов должно быть от 0 до 23. Для одной таблицы может быть только одно правило.
Представление	Представление есть логический способ просмотра данных в одной или нескольких таблицах базы данных. Представление является виртуальной таблицей; физически она не присутствует в базе данных. Представление содержит указатели на существующие в базе данных таблицы. Представления могут использоваться для составления подмножества информации, хранимой в одной или нескольких таблицах, если не используется иная информация. Это особенно полезно, если некоторые столбцы таблицы содержат чувствительную информацию. Запросы к представлениям выглядят как запросы к обычной физической таблице. Представления входят в стандартные функции сервера SQL. В IndustrialSQL Server, тем не менее, к данным можно обращаться при помощи таблиц расширения, которые отличаются от обычных представлений. См. также таблица расширения.
Приоритет	Приоритет событий определяет, как будут выполняться события, если система перегружена и не может обработать все события. События, которые имеют приоритет "критические", будут выполняться перед событиями с приоритетом "нормальные".

---

Протокол передачи гипертекста (НТТР)	НТТР есть протокол передачи информации через Интернет.
Псевдоним	Псевдоним есть имя, под которым сетевой сервер известен сетевым клиентам. Сервер может иметь несколько псевдонимов.
Путь хранения	Путь хранения есть путь к каталогу, в который сохраняются файлы архивных данных.
Разнородный запрос	Разнородным называется запрос, обращающийся к данным из различных непохожих источников данных.
Распределенная модель составного объекта	Распределенная модель составного объекта DCOM есть протокол, позволяющий компонентам программного обеспечения напрямую взаимодействовать через сеть.
Распределенная модель составного объекта (DCOM)	
Реальное время	Операции реального времени совершаются с той же скоростью, что и физические процессы. В среде реального времени компьютер должен реагировать на ситуации по мере их возникновения. Эти ситуации могут включать в себя сдвиг переключателя или включение печи.
Результат	Результатом называются характеристики или атрибуты любого объекта, выявленные запросом к базе данных.
Реляционная база данных	Реляционная база данных есть такое устройство базы данных, которое упорядочивает данные в соответствии с отношениями между ними. В реляционной базе данных отношения между элементами данных выражаются посредством таблиц. Например, могут подаваться запросы на поиск данных в одной таблице плюс в любых связанных с ней таблицах.
Сбор данных	Процесс получения переменных данных из процессов производственного участка.
Сброс	Сброс есть процесс удаления базы данных или устройства базы данных.
Сводка	Операция сводки (MIN, MAX, SUM, AVG) тэга, запланированная пользователем и автоматически выполняемая системой управления событиями в соответствии с расписанием.

---

Сводные данные	Сводные данные есть данные, полученные в результате внутренних подсчетов, проводимых IndustrialSQL Server (максимум, среднее, сумма и т.п.). Сводные данные обеспечивают верхний уровень просмотра данных и позволяют использовать меньше пространства для данных, которые требуется хранить долгое время. Например, среднее значение пяти тэгов.
Связанный сервер	Связанный сервер есть поставщик данных OLE DB, связанный с сервером SQL. См. также <i>поставщик данных OLE DB</i> .
Связка	Связка есть класс запроса SQL, который запрашивает данные из одного или нескольких столбцов двух или большего числа таблиц.
Сервер DDE	Сервер DDE есть приложение, поставляющее данные клиенту DDE через протокол Microsoft Dynamic Data Exchange. См. также Динамический обмен данными.
Сервер базы данных IndustrialSQL	Сервер базы данных IndustrialSQL (PDSSRV.EXE) перехватывает все запросы клиента на данные до того, как они достигнут Microsoft SQL Server, а если запрос идет на извлечение данных из архивных блоков IndustrialSQL Server, сервер базы данных контактирует с системой получения IndustrialSQL Server, получая данные и представляя их пользователю в виде нормальной таблицы Microsoft SQL Server. Запросы, не касающиеся архивных блоков, неизменными проходят в Microsoft SQL Server для стандартной обработки.
Сервер PB	Сервер реального времени (PB) есть компонент системы IndustrialSQL Server, принимающий значения данных от устройств сбора производственных данных, проводящий все необходимые операции и отправляющий значения в систему хранения. См. также система хранения
Сервер реального времени	См. Сервер PB.
Символ	(1) символом называется столбец, служащий для идентификации строки. Символ строки должен быть уникален в таблице. (2) Символ есть папка в реестре Windows NT. Символы похожи на заголовки разделов в файле .INI.

---

Символов в секунду (CPS)	Количество символов в секунду есть общее число символов, посланных в систему хранения всеми источниками данных. Источники данных включают в себя серверы ввода-вывода, IndustrialSQL RT Server и системные драйверы. Количество символов в секунду может быть просмотрено при помощи управляющего приложения IndustrialSQL Server.
Синхронизация времени	Синхронизация времени есть механизм, с помощью которого IndustrialSQL Server посылает сообщение в серверы ввода-вывода для синхронизации штампов даты/времени серверов ввода-вывода с временем IndustrialSQL Server.
Система хранения	Система хранения IndustrialSQL Server отвечает за обработку (если требуется), архивирование и хранение данных на диске.
Системное именованное пространство	Системное именованное пространство есть созданный системой набор компонентов системы IndustrialSQL Server. Системное именованное пространство содержит объекты, составляющие систему IndustrialSQL Server, такие как серверы ввода-вывода, узлы и т.п. IndustrialSQL Server заполняет системное именованное пространство на основе текущей конфигурации производственного участка, которая хранится в базе данных в таблицах системной конфигурации. Иерархия содержимого системного именованного пространства может быть просмотрена в дереве каталогов. См. также именованное пространство, общее именованное пространство, личное именованное пространство, браузер, хранимые процедуры.
Системные тэги	Системным тэгом называется заранее определенная переменная системы. Системные тэги InTouch имеют префикс \$. Например, \$DateTime. Системные тэги IndustrialSQL имеют префикс SYS. Например, SysTimeSec.
Системный администратор (SA)	Системным администратором называется человек, ответственный за администрирование и сопровождение сервера SQL. Функции администрирования и сопровождения включают в себя изменение базы данных, администрирование безопасности базы данных, резервирование данных и т.п.
Скорость изменения	Скоростью изменения называется скорость изменения значения тэга за определенный период времени, обычно выраженная в процентах.
Скорость опроса	Скорость опроса есть скорость, с которой данные считываются из устройства сбора.

---

Скорость хранения	Скорость хранения есть интервал времени, с которым периодически сохраняются значения тэгов.
Словарь данных	Словарь данных есть группа таблиц, содержащих информацию обо всех объектах базы данных.
Сложный тэг	Сложный тэг содержит данные, которые не могут быть отнесены ни к одной из четырех основных категорий типов тэгов: аналоговых, дискретных, текстовых или тэгов событий. Сложные тэги включают в себя BLOВЫ, такие как массивы, файлы с расширением AVI, рисунки и т.п. <i>См. также</i> крупный бинарный объект.
Службы открытых данных (ODS)	Службы открытых данных (ODS) есть интерфейс программирования приложений (API) между сетевыми библиотеками сервера и приложениями сервера. Службы ODS позволяют приложениям-клинтам получать доступ к данным, которые не хранятся физически на сервере Microsoft SQL, к которому они присоединяются.
Снимок	Снимок есть набор значений тэгов данных в конкретный момент времени. Когда в системе происходит событие, значения всех аналоговых или дискретных тэгов в этот момент могут быть зафиксированы и сохранены. Данные снимка используются для определения состояния производственной среды в момент происхождения определенного события в системе.
Событие	Событие есть проявление определенной активности в системе. Определение события сохраняется как тэг события. События засекаются детекторами событий и с ними можно производить действия. <i>См. также</i> детектор, действие, тэг события.
Соединение открытых баз данных (ODBC) Microsoft	ODBC есть API-интерфес соединения, независимый от баз данных .
Составной индекс	Составной индекс есть индекс, использующий от 2 до 16 столбцов как первичный ключ. Например, составной индекс может использовать и <i>дату/время</i> и <i>имя тэга</i> в составном индексе из 2 столбцов.
Составной символ	Составной символ есть символ, который составляется из двух или большего числа столбцов таблицы. Составные столбцы требуют большего количества сложных связей.

Список действий	Список действий системы управления событиями есть пространство в потоке действий, где происходит назначение порядка действий для выполнения. Действие над событием может быть или в критическом, или нормальном списке действий. У каждого тэга есть свой список.
Среда выполнения	Средой выполнения называется период, когда данные переносятся модулем управления и происходит их обработка арифметическо-логическим модулем. Также это период выполнения программы.
Строка	В таблице строкой называется набор стоящих рядом столбцов информации, описывающих определенную ячейку базы данных. Например, для ячейки "личность" строка может содержать столбцы, описывающие рост, вес, цвет волос, возраст человека и т.п.
Сценарий	Сценарием называется набор выражений SQL, используемый для проведения действий над базой данных, таких как изменение данных или добавление новых объектов. Сценарии могут сохраняться как хранимые процедуры или файлы.
Счетчик пакетов	В IndustrialSQL Server количество пакетов данных, полученных системой хранения.
Счетчик строк	Расширения временного домена счетчика строк есть функция IndustrialSQL Server, которая при циклической выборке отличается от обычного счетчика строк сервера SQL. Применение счетчика строк расширения временного домена зависит от того, используется ли циклическая или же дельта-выборка и запрашивается ли широкая таблица.
Таблица	Таблицей называется группа стоящих рядом ячеек базы данных и их характеристик. См. также строка, столбец.

---

Таблица расширения	В IndustrialSQL Server таблицей расширения называется таблица, позволяющая обрабатывать полученные производственные данные. Эти таблицы не являются частью обычных функций SQL Server. Стандартные таблицы SQL Server хранят данные, которые непосредственно и физически расположены в устройстве данных. Таблица расширения, тем не менее, представляет данные как реально существующую таблицу, но таблица эта не существует в устройстве данных. Таблица расширения есть логическая таблица, заполненная данными из файлов, созданных системой хранения. Данные в таблице расширения управляются с помощью обычного кода Transact-SQL так же, как и специализированные расширения SQL, поставляемые с IndustrialSQL Server и называемые Industrial-SQL. <i>См. также</i> представление, таблица конфигурации.
Таблицы конфигурации	Таблицы конфигурации содержат информацию, определяющую основные аспекты работающей системы. Например, в таблицах конфигурации содержатся определения тэгов, серверов ввода-вывода, пользователей и т.п. <i>См. также</i> таблицы расширения.
Текстовое значение	Текстовым значением называется текстовое выражение, представленное как одиночный элемент данных. Текстовые значения не требуют специального синтаксиса.
Тема	В протоколах DDE или SuiteLink тема есть определяемая приложением подгруппа элементов данных. Эти протоколы используют формат наименования из трех частей для поиска информации в приложениях. Для того, чтобы приложение (например, InTouch) получало данные, ему должны быть известны имя приложения, тема и элемент. Образцом темы может служить имя электронной таблицы Excel. <i>См. также</i> Динамический обмен данными, SuiteLink.
Тип данных	Тип данных указывает, какой тип информации может содержать столбец таблицы и как эта информация сохраняется. Есть два основных типа данных: системные и пользовательские.

---

Транзакция	Транзакцией называется набор из одного или более командных сценариев, читающих и/или записывающих в базу данных. Транзакция есть запрос в систему IndustrialSQL Server на поиск, ввод, изменение или выдачу информации об объекте реляционной базы данных. Все транзакции выполняются во время запуска как одна рабочая единица. Если в одном из сценариев происходит ошибка, вся транзакция будет возвращена к исходному состоянию базы данных до тех пор, пока начатая транзакция не будет завершена.
Триггер	Триггером называется пакет кодов, привязанный к таблице базы данных. Триггер срабатывает при изменении данных в таблице. Триггеры используются для переноса изменений из одной таблицы базы данных в другую с целью обеспечения целостности. <i>См. также</i> целостность связей.
Тэг	Тэг определяется как элементарная переменная аналогового, дискретного, текстового или сложного типа, которая хранится в базе данных IndustrialSQL Server. Проще говоря, тэг есть средство связи с механизмом или устройством производственного участка. Также он может быть связан с системными переменными, такими как системное время (SYSTIMESEC).
Тэг события	Тэгом события называется определение события в системе. Например, если требуется зарегистрировать событие, когда температура танкера достигает 100 градусов, следует определить тэг события и назвать его "TankAt100". <i>См. также</i> детектор, действие, тэг события.
Тэги DDE	Тэг DDE есть тэг, значения которого читаются или записываются из или в другое приложение через протокол Microsoft Dynamic Data Exchange. <i>См. также</i> Динамический обмен данными.
Удаленная таблица	Удаленной называется таблица, представляющая данные для нелокального хранилища данных, доступ к которому осуществляется посредством OLE DB. <i>См. также хранилище данных, Object Linking and Embedding for Databases, таблица расширения.</i>
Уникальный индекс	Уникальным называется индекс, в котором нет двух строк таблицы с дублируемым значением. <i>См. также</i> индекс.
Устройство	<i>См.</i> устройство базы данных, устройство дампа, устройство журналов.

---

Устройство базы данных	Устройство базы данных есть файл, в котором хранятся базы данных. Одна база данных может храниться на нескольких устройствах.
Устройство дампа	Устройство дампа есть файл, который сохраняет резервную копию базы данных. <i>См. также</i> резервирование.
Устройство журнала	Устройством журнала называется файл, содержащий журнал операций с базой данных. <i>См. также</i> журнал действий.
Учетная запись	Учетная запись, или идентификатор пользователя, есть уникальное имя, используемое пользователем базы данных для входа в систему. Учетная запись может содержать до 30 буквенных или числовых символов, где первым символом должна быть буква или символы # или _.
Хранение методом исключения	<i>См.</i> дельта-хранение.
Хранилище данных	Хранилище данных есть файл, содержащий данные. Нелокальное хранилище данных есть просто хранилище данных, существующее вне базы данных Microsoft SQL Server.
Хранимая процедура	Хранимая процедура — это скомпилированная группа выражений SQL. Хранимые процедуры позволяют выполнять последовательные действия при помощи выражения SQL. Обычно хранимая процедура вызывается другой программой для выполнения; автоматически, в ответ на событие, она не выполняется. Хранимые процедуры могут использоваться в качестве ярлыков для часто используемых наборов выражений SQL, обеспечивающих дополнительную функциональность. Пользователи IndustrialSQL Server могут использовать любые хранимые процедуры системы Microsoft SQL Server, а также хранимые процедуры системы IndustrialSQL Server. Хранимые процедуры IndustrialSQL Server начинаются с ww_. Также поддерживаются пользовательские хранимые процедуры.
Целое	Целое есть любой член набора данных, состоящий из положительных, отрицательных чисел и нуля. Примеры: -59, -3, 0.
Целостность данных	Целостность данных есть совместимость и точность данных, хранимых в базе данных.

Целостность связей (RI)	Целостность связей есть механизм, удостоверяющий, что все внешние ключи имеют связанный первичный ключ во взаимодействующих таблицах. Целостность связей предотвращает указание внешних ключей на несуществующие первичные ключи и ускоряет связку таблиц. См. также первичный ключ, внешний ключ.
Циклическая выборка	Циклическая выборка есть получение сохраненных данных за заданный период времени, основанное на указанной дискретности, независимо от того, изменилось ли значение тэга (тэгов). См. также дельта-выборка, дискретность.
Циклическое хранение	Циклическое хранение есть процесс хранения данных, основанный на интервале времени. Например, можно сохранять значение аналогового тэга каждую секунду. Циклическое хранение используется для сохранения аналоговых значений, физические свойства которых производят изменения данных быстрее, чем скорость сохранения. См. также дельта-хранение.
Циклическая область хранения	Циклическая область хранения есть каталог на компьютере, используемый для хранения файлов архивных данных.
Шаблонный символ	Шаблонный символ есть клавиша, используемая для обозначения одного или нескольких символов. При поиске в базе данных сервера SQL используйте знак подчеркивания (_), знак процента (%), и скобки ([ ]) с ключевым словом LIKE, с которым должна совпадать выборка в базе данных. Например, для поиска всех тэгов в системе, начинающихся на "sys", задайте поиск "sys%".
Широкие таблицы	В IndustrialSQL Server широкие таблицы показывают значения одного или нескольких тэгов за период времени. Каждая строка содержит штамп даты/времени для данных и значений одного или нескольких тэгов, указанных в запросе. См. также <i>живые таблицы</i> , <i>архивные таблицы</i> .
Элемент	В протоколах DDE или SuiteLink элементом называется "заглушка" значения данных. Протокол DDE использует формат наименования из трех частей для нахождения информации между приложениями. Для того, чтобы приложение (такое как InTouch) получало данные, оно должно знать имя приложения, тему и элемент. Примером элемента служит имя ячейки в электронной таблице Excel. Другим примером может служить имя тэга в InTouch. См. также Динамический обмен данными, SuiteLink, тема.

Язык  
структурированных  
запросов (SQL)

SQL есть язык, используемый в системах  
реляционных баз данных для определения, поиска и  
управления данными.

# Предметный указатель

## A

ActiveDataGrid, 1-16  
 ActiveEvent, 7-4  
     Атрибуты безопасности, 7-42  
 ActiveGraph, 1-16  
 ActiveTagBrowser, 1-16  
 ActiveTimeSelector, 1-16  
 ActiveTrend, 1-16  
 AltData, 4-8  
 ANASTOR.EXE, 4-16, 5-8

## B

binary, тип столбца, 2-47  
 bit, тип столбца, 2-47  
 BufferDisk, 4-8

## C

char, тип столбца, 2-47  
 ConnectTo, 11-4, 11-6  
 CPS, 8-2

## D

DatabaseVersion, 2-19  
 datetime, тип столбца, 2-47  
 DAT-кассета, 4-19  
 DBMSSOCN, 12-15  
 DCOMCNFG.EXE, 7-42  
 DDE, 2-43, 10-2, 10-5  
 decimal, тип столбца, 2-47  
 DIGSTOR.EXE, 4-16, 5-8  
 DynData, 4-8

## E

EL\_DEBUG, 14-3  
 EL\_DISPLAY, 14-3  
 EL\_INFO, 14-3  
 EL\_WARNING, 14-3  
 error logging, A-5  
 Event Viewer, 8-10  
 EventStorageDuration, 2-19  
 EVENTSYS.EXE, 4-17, 5-8

## F

FactorySuite, A-12  
 FastDDE Серверы ввода-вывода, 1-8

## I

Industrial Workbook, 1-15  
 IndustrialSQL Server  
     Доступ к Microsoft SQL Server, 5-18  
     Загрузка ресурсов сервера, 8-9

    Запуск, 5-7  
     Интеграция с Microsoft SQL Server, 11-2  
     как архивного провайдера, 3-14  
     Лицензия, 5-2  
     Настройка протоколов, 11-6  
     Обзор, 1-2  
     Оценка производительности, 8-2  
     Перевод часов на зимнее/летнее время, 4-20  
     Переменные целостности, 8-3  
     Повторная инициализация, 2-44  
     Псевдонимы, 11-5  
     Регистрация, 11-10  
     Связь с InTouch SQL Access Manager, 3-15  
     Сетевое соединение, 11-7  
     Синхронизация метки времени, 10-13  
     Целостность системы, 8-6  
 InSQL Configure, 1-14  
     Главное окно, 2-5  
     Запуск, 2-3  
     Описание, 2-2  
     Панели инструментов, 2-6  
     Панель свойств, 2-12  
 InSQL Control, 1-14  
     Описание, 5-2  
     Панель инструментов, 5-4  
     Повторная инициализация, 2-46  
     Строка состояния, 5-4, 5-5  
 InSQL QuickLook, 1-15  
 InSQL Report, 1-15  
 InSQL Trend, 1-15  
 InSQL VectorMaster, 1-15  
 InSQL VectorViewer, 1-15  
 INSQLIOS.EXE, 4-16, 5-8  
 InSQLListenOn, 2-19  
 integer, тип столбца, 2-47  
 InTouch  
     Перенаправление сервера ввода-вывода, 10-5  
 InTouch, 2-2  
 InTouch SQL Access Manager, 3-15  
 IODRIVER.EXE, 4-16, 5-8  
 IPX/SPX, 11-7  
     Настройка, 12-11  
 IP-адрес, 12-15

## K

Key, 11-5

## L

LAN, 1-7  
 ListenOn, 11-4, A-11

## M

Microsoft Access, 12-19  
 Microsoft Excel, 12-19  
 Microsoft SQL Server  
     Безопасность, 13-2  
     Запуск, 11-8  
     Интеграция с IndustrialSQL Server, 11-2  
     Лицензия, 5-2  
     Сетевые соединения, 11-4  
 Microsoft Word, 12-19  
 ModLogTrackingStatus, 2-19

MSFMERF.EXE, 4-16, 5-8

Multi-protocol, 11-7

Настройка, 12-13

Настройка клиента, 12-13

Соединение, 11-6

## N

Named pipes, 11-6, 11-7

Настройка, 12-9

Named pipes, 12-2

NetBIOS, 12-9

NetWare, 12-11, 12-13

NetWare IPX/SPX

Настройка клиента, 12-11

NWLink, 12-13

NWLink IPX/SPX, 11-6

## O

Object Linking and Embedding for Databases, 6-4

ODBC, 1-15, 3-15, 11-7

Настройка клиента, 12-19

OLE DB, 6-4

## P

PDSMGR.EXE, 4-16, 5-8

PDSSRV.EXE, 4-16, 5-8

## R

RDBMS, 11-2

Ограничения, 1-3

real, тип столбца, 2-47

REGEDT32.EXE, A-2

RPC, 12-13

RTSERVER.EXE, 4-16, 5-8

## S

SQL, 1-3

SQL Server

таблицы, 4-7

SQL Server Client Network Utility, 1-14

SQL Server Enterprise Manager, 1-14, 11-11, 13-4

Использование, 11-10

SQL Server Service Manager, 1-14

SQLListenOn, 2-19

StringStorageDuration, 2-19

STRSTOR.EXE, 4-16, 5-8

SuiteLink, 2-43, 10-2, 10-5

SummaryStorageDuration, 2-19

SYSDRV.EXE, 4-16, 5-8

SystemVersion, 2-19

## T

Tag Finder, 7-39

TCP/IP, 11-7

Настройка, 12-15

TimeDeadband, 2-45

Transact-SQL, 1-5, 1-8

## V

ValueDeadband, 2-45

## W

WAN, 1-7

WORM, 1-10

WormDisk, 4-8

ww\_InSQLStatusSelect, хранимая процедура, 2-46

ww\_InSQLStatusSet, хранимая процедура, 2-46

wwAdmin, 13-6, 13-11

wwdbo, 13-6, 13-11

wwPower, 13-6, 13-11

wwUser, 13-6, 13-11

## X

xp\_DiskCopy, 4-12, 9-3

xp\_NewHistoryBlock, 5-16

xp\_proclist, 9-3

## A

Автоматический запуск

Microsoft SQL Server, 11-9

Адрес IPX, 12-12

Адрес транспортировки IPX, 12-11

Адреса гнезд TCP/IP, 12-15

Адресная строка

Редактировать для тэга, 2-22

Активный образ, 4-3, 5-8

Альтернативная область хранения, 4-8, 4-15, A-3

Описание, 4-12

Аналоговое хранение, 5-4

Аналоговый тэг

Добавить, 2-29

Настройка, 2-24

Настройка методов хранения, 2-26

Повторная инициализация, 2-44

Редактировать общие свойства, 2-24

Редактировать свойства хранения данных, 2-26

Фиксация событий, 7-4

Архив сводок, 2-49, 7-34

Архив событий, 2-49, 7-9, 7-47

Архив текстовых данных, 2-49

Архивные блоки, 5-7, 6-4, 9-2, A-10

Каталог, 4-9

Копирование, 9-3

Место хранения, 4-8

Нотация папки, 4-9

Перезапись, 4-14

Просмотр, 4-10

Просмотр в InSQL Configure, 2-15

Размер, 4-9

Создание, 5-7

Хранение, 4-7

Архивные данные, 1-2, 1-10

Выборка, 9-2

Резервное копирование, 9-2

Хранение, 4-7

Архивный блок

Запуск нового, 5-16

Архитектура, 1-7

**Б**

- База данных
  - Безопасность, 13-3
  - Стандартные учетные записи, 13-6
- База данных Runtime, 1-10, 2-14, 3-2, А-8
  - Резервирование, 3-5
  - Номер версии, 2-14
- База данных Holding, 2-14
- Базы данных
  - Восстановление, 9-7
  - Резервные копии, 9-4
  - Управление, 11-13
- Безопасность
  - SQL Server, 13-2
  - База данных, 13-3
  - Выражение, 13-3
  - Объект, 13-3
  - Операционная система, 13-2
  - Уровни, 13-2
- Библиотека БД, 11-7
  - Клиенты, 12-9
  - Описание, 12-5
- Библиотека ДБ
  - Параметры, 12-8
- Бизнес-данные, 1-2
- Блок данных. См. Архивные блоки
- Бройзер
  - Фильтр, 2-9
- Броузер
  - Показать/скрыть, 2-12
- Буферная область хранения, 4-8, А-3
  - Описание, 4-12

**В**

- Внешний детектор
  - настройка, 7-19
  - Описание, 7-4
- Внутренний действительный тэг, 3-10
- Внутренний дискретный тэг, 3-10
- Внутренний текстовый тэг, 3-10
- Внутренний целый тэг, 3-10
- Временное смещение, А-8
- Выборка
  - Архивные данные, 9-2
  - Резервные копии, 9-2
- Выборка данных, 1-11, А-8
  - ODS и OLE DB, 6-8
  - Активный образ, 4-4
  - Дельта, 6-2
  - Мертвая зона времени, 2-27
  - Мертвая зона значения, 2-27
  - циклическое, 6-2
- Выборки, 4-3
- Выражения, 13-3
  - Установка прав доступа, 13-18

**Г**

- Глобальная сеть, 1-7
- Гнезда TCP/IP, 11-6
- Группы, 2-8, 13-3, 13-11
  - Добавление, 13-14

**Д**

- Дамп, 9-4
- Данные временной последовательности, 1-5, 1-11
- Данные конфигурации, 1-2
  - Хранение, 4-5
- Данные реального времени, 1-2, 1-8
- Данные событий, 1-2
  - Выборка, 7-47
- Действие над общим SQL
  - Описание, 7-5
- Действие над снимком
  - Описание, 7-5
- Действие над событием
  - Выделение системного потока, 7-7
  - Изменение мертвой зоны, 7-33
  - Мертвая зона, 7-5
  - Общее действие SQL, 7-26
  - Общий SQL, 7-5
  - Отправка e-mail, 7-5
  - Сводка, 7-5
  - Снимок, 7-5, 7-28
  - Создание сводки, 7-34
  - Электронное сообщение, 7-29
- Действие с мертвой зоной
  - Описание, 7-5
- Действие с электронной почтой
  - Описание, 7-5
- Действие со сводкой
  - Описание, 7-5
- Действия над событием, 7-7
- Дельта-выборка, 1-11, 6-2
- Дельта-реимпортирование, 3-4
- Дельта-хранение, 1-9, 2-27
  - Изменение метки времени, 10-15
  - Оптимизация, 4-23
  - Скорость, 2-27
- Дельта-Хранение данных
  - Описание, 4-2
- Детектор планировщика
  - Настройка, 7-24
- Детектор событий, 7-4
  - Внешний, 7-4, 7-19
  - Выделение системного потока, 7-6
  - Детектор планировщика, 7-24
  - Общий SQL, 7-4
  - Общий детектор SQL, 7-20
  - Определение края, 7-8
  - Планировщик, 7-4
  - Указанное аналоговое значение, 7-4
- Детектор события
  - Детектор указанного аналогового значения, 7-22
  - Детектор указанного дискретного значения, 7-23
  - Задержка, 7-6
  - Детектор указанного аналогового значения
    - Настройка, 7-22
    - Описание, 7-4
  - Детектор указанного дискретного значения
    - Настройка, 7-23
    - Описание, 7-4
- Диалоговое окно About, 2-13
- Диски WORM, 4-12

Дискретное хранение, 5-4  
 Дискретный тэг  
   Добавить, 2-35  
   Настройка, 2-33  
   Повторная инициализация, 2-44  
   Редактировать общие свойства, 2-33  
   Фиксация событий, 7-4  
 Диспетчер дискового пространства, 5-8  
 Длина текстового тэга, 2-39  
 Допуск, 13-3  
 Доступ на уровне сервера, 13-2  
 Драйвер DDE, 5-4  
 Драйвер ввода-вывода  
   Сообщения об ошибках, 14-7  
 Драйвер мониторинга системы, 1-13  
 Драйвер последовательного порта, 5-4  
 Драйвер состояния системы, 5-8  
 Драйвер устройства  
   Настройки реестра, 10-19  
 Драйверы данных, 5-13  
 Дублируемые имена тэгов, 3-9

## Е

Единица измерения  
   Добавить, 2-32  
   Настройка, 2-25, 2-30  
   Повторная инициализация, 2-45  
   Просмотр набора для сервера, 2-30  
   Просмотр текущих для сервера, 2-30  
   Редактировать, 2-30

## Ж

Журнал ошибок  
   Категории, 14-2  
   Расположение, 14-3, 14-6  
   Управление, 14-2  
 Журнал событий, 8-10, 14-3, 14-4, 14-5  
 Журналы транзакций, А-9

## З

Загрузка ресурсов сервера, 8-9  
 Запросы, 12-11  
 Запуск  
   Автоматический, 5-9  
 Значение качества, А-10  
 Значение Реестра ListenOn, 11-5  
 Значение с плавающей точкой, 2-22

## И

Идентификатор пользователя, 13-2, 13-6, 13-21  
   Добавление, 13-8  
   Просмотр, 13-6  
   Управление, 13-6  
 Идентификатор процесса, 12-9, 12-15  
 Идентификатор узла, 11-4, 12-9, 12-15  
 Изменение базы данных, 2-46  
   Отслеживание, 2-47  
   Таблицы отслеживания, 2-47  
 Изменение мертвой зоны  
   Настройка, 7-33  
 Именной системный параметр

  Просмотр в InSQL Configure, 2-19  
 Именные пространства, 1-14, 4-5  
 Именные системные параметры  
   Просмотр, 2-19  
 Импорт, 3-2  
 Импорт тэгов, 2-2  
   Дельта-реимпортирование, 3-4  
   Дублированные имена тэгов, 3-3  
   Импортирование из InTouch, 3-6  
   Объяснение, 3-2  
   Порядок импортирования, 3-3  
   Редактирование имен компьютеров, 3-4  
 Импортирование  
   Дельта-реимпортирование, 3-4  
   Дублированные имена тэгов, 3-3  
   Порядок импортирования, 3-3  
   Редактирование имен компьютеров, 3-4  
 Имя DLL, 12-15  
 Имя канала, 12-9  
 Имя компьютера, 3-4, 10-2, 12-11, 12-13  
 Имя пользователя, 13-8, 13-11  
 Имя сервера, А-9  
 Имя тэга  
   Дублирование, 3-3  
 Имя устройства, 10-20  
 Имя элемента, 2-31  
   Редактировать для тэга, 2-22  
 Индикаторы состояния, 5-7  
 Инструменты администратора, 1-14  
 Интервал сохранения, 4-2  
 Информационные сообщения, 14-2  
 Информация сводок  
   Просмотр в InSQL Configure, 2-16  
 Источник данных ODBC, 12-19  
 Источники данных, 6-4, А-10

## К

Клиент TCP/IP  
   Настройка, 12-15  
 Клиент/сервер, 11-7  
 Клиентские приложения, 1-7, А-12  
 Ключ Datapaths, 4-8  
 Ключ Setup, А-9  
 Коммуникационные приложения, 1-8  
 Копирование  
   архивные блоки, 9-3  
 Критические ошибки, 14-2  
 Кэш диска, 8-9

## Л

Линейная интерполяция, 2-23  
 Локальная сеть, 1-7

## М

Магнитная лента, 9-2  
 Максимальное значение, 2-25  
 Масштабирование, 2-23  
 Мертвая зона, 1-9  
 Мертвая зона времени, 1-9, 2-27  
 Мертвая зона значения, 1-9, 2-27  
 Мертвая зона хранения  
   Действие над событием, 7-5

Место хранения, 4-8  
 Изменение, 4-8  
 Повторная инициализация, 2-44  
 Редактирование, 2-42  
 Метка качества, 1-8  
 Механизмы IPC, 12-13  
 Минимальное значение, 2-25  
 Минимальный порог, 4-11, 4-14  
 Монитор производительности, 1-15  
 Мониторинг  
 Драйверы обмена данными, 8-2  
 Обмен данными, 8-2  
 Переменные целостности, 8-3  
 Мониторинг системы, 1-8

## Н

Настройка  
 Соединения клиента, 12-2, 12-8  
 Настройка драйвера, 10-19, 10-20  
 Необработанное значение  
 Редактировать для тэга, 2-22  
 Непрерывный запрос, 1-11  
 Номер гнезда TCP/IP, 12-15  
*Номер порта*, 12-15

## О

Обзор, 1-2  
 Области буферного хранения, 1-10  
 Области хранения, 1-9  
 Альтернативная, 4-12  
 Буферная, 4-12  
 Постоянная, 4-12  
 Циклическая, 4-11  
 Область альтернативного хранения  
 Редактировать путь, 2-42  
 Область буферного хранения  
 Редактировать путь, 2-42  
 Область постоянного хранения  
 Редактировать путь, 2-42  
 Область циклического хранения  
 Редактировать путь, 2-42  
 Обмен данными  
 Мониторинг драйверов, 8-2  
 Обработчик данных, А-5  
 Обработчик сообщений, 5-8  
 Общее действие SQL  
 Настройка, 7-26  
 общие группы, 2-8  
 Общие ресурсы  
 Настройка, 3-5  
 Общий детектор SQL  
 Настройка, 7-20  
 Описание, 7-4  
 Общий счетчик пакетов, 5-11  
 Объект  
 Безопасность, 13-3  
 Объект базы данных  
 Управление, 11-14  
 Объекты базы данных, 1-12, 13-3  
 Операционная система  
 Безопасность, 13-2  
 Операция сводки, 2-17  
 Добавить аналоговый тэг, 2-28

Добавление, 7-36  
 Определение края, 7-8  
 Оптические устройства, 4-19  
 Отладочные сообщения, 14-2  
 Отслеживание  
 Драйверы сбора данных, 5-11  
 Сбор данных, 5-12, 5-13  
 Система сбора данных, 5-2  
 Система хранения данных, 5-2  
 Счетчик пакетов, 5-11  
 Отслеживание изменений, 2-47. *См. также*  
 Изменение баз данных  
 Ограничения, 2-47  
 Таблицы, 2-47  
 Ошибки, 5-5  
 Драйвер ввода-вывода, 14-7  
 Сброс счетчика, 5-16

## П

Пакетный сбор, 5-4  
 Память, 4-3  
 Панели инструментов  
 InSQL Configure, 2-6  
 InSQL Control, 5-4  
 Показать/скрыть, 2-12  
 Панель свойств, 2-12  
 Пара сообщений, 2-34  
 Добавить, 2-38  
 Настройка, 2-36  
 Повторная инициализация, 2-45  
 Просмотр текущих для сервера, 2-36  
 Редактировать, 2-36  
 Парадигмы хранения, 4-2  
 Пароли  
 Идентификатор пользователя, 13-8  
 Стандартные, 13-11  
 Управление, 13-19  
 Перевод часов на зимнее/летнее время, 4-20  
 Передача данных, 5-12  
 Переменные состояния, 4-16, 5-8, 5-11, 8-3  
 Запрос, 8-6  
 Планировщик событий  
 Описание, 7-4  
 Повторная инициализация, 1-9  
 Описание, 2-44  
 После импортирования, 3-13  
 Хранимые процедуры, 2-46  
 Повторное импортирование, 3-13  
 Поиск тэгов, 7-39  
 Получатель данных, 5-4  
 Полученное необработанное значение, 2-23  
 Пользователи  
 Добавление, 13-13  
 Управление, 13-11  
 Пользователи базы данных. *См. Пользователь*  
 Пользовательские тэги, 3-2  
 Порог, 2-43, 4-11, 4-14, А-3  
 Поставщик архива, 3-14  
 Поставщик данных OLE DB  
 Описание, 6-4  
 Постоянная область хранения, 4-8, А-3, А-4  
 Описание, 4-12  
 Поток, 7-6  
 Права допуска

- Управление, 13-15
  - Права допуска к объекту
    - Установка, 13-15
  - Предел, 2-17
  - Предупреждения, 14-2
  - Привилегии, 1-12
  - Приемник данных, А-5
  - Приложение InTouch, 2-2, 3-4, 3-6, 3-7
  - Проводник Windows, 1-14
  - Программные элементы
    - Отобразить/скрыть, 5-6
    - Показать/скрыть, 2-12
  - Производительность системы, 8-7
  - Просмотр
    - Сообщения состояния, 5-14
    - Темы, 5-14
  - Просмотр лицензии, 2-13
  - Просмотр событий, 1-15
  - Протокол IPC, 12-13
  - Протоколы, 10-5
  - Процессы, 4-16, 9-7, 12-2
  - Псевдоним
    - IndustrialSQL Server, 11-5
    - Создание, 12-4, 12-7
  - Пути, 2-42
- Р**
- Размер диска, 4-13
  - Разнородный запрос, 6-4
  - Разрешение выборки, 1-2, 7-36
  - Разрешение имен, 12-14
  - Разрешение подсчета, 7-36
  - Расширения временного домена, 6-3, 6-6
  - Регистрация ошибок, 14-2, А-13. См. Журнал ошибок
  - Редактирование
    - Реестр, А-2
  - Редактировать
    - Место хранения, 2-42
  - Редактор реестра
    - запуск, 11-3
    - Редактирование, А-2
  - Редактор реестра Windows, 1-14
  - Режим безопасности, 13-4
  - Режимы просмотра
    - Performance Monitor, 8-8
  - Резервирование, 1-8, 1-10
  - Резервное копирование
    - Архивные данные, 9-2
    - Выполнение, 9-2
  - Резервные копии, 3-5, 4-19, 9-4, 9-5
    - Базы данных, 9-4
    - Выборка данных, 9-2
    - Создание, 9-4
  - Реляционная система управления базой данных (RDBMS)
    - Ограничения, 1-3
- С**
- Сбор данных, А-5
    - Драйверы отслеживания, 5-11
    - Отслеживание, 5-12, 5-13
    - Редактирование конфигурации, 2-21
  - Уровни, 5-8
  - Сводка, 1-6, 1-8
    - Настройка, 2-28
  - Сводные данные, 1-2
  - Сводный расчет
    - Конечное время, 2-18
    - Начальное время, 2-18
  - Свойства хранения данных
    - Редактировать, 2-26
  - Связанный сервер, 6-7
  - Сервер
    - Просмотр свойств, 2-14
  - Сервер базы данных, 5-4, 6-2
  - Сервер ввода-вывода, 4-5
  - Сервер ввода-вывода InSQL, 2-43, 5-4, 5-8, 10-2, 10-18, А-5
  - Сервер ввода-вывода реального времени, 10-18
  - Сервер выборки, А-5
  - Сервер сбора данных, 5-7
  - Сервер тэгов, 3-3
  - Серверные службы, 1-7
  - Серверы SuiteLink, 1-8
  - Серверы ввода-вывода, 1-8, 1-9, 5-11, 8-2
    - Добавление, 10-6
    - Импорт конфигурации, 3-5
    - Настройка, 2-43, 10-2
    - Настройка через InSQL Configure, 10-4
    - Настройка через InTouch, 10-3
    - Настройки реестра, 10-19
    - Перенаправление в InTouch, 10-5
    - Повторная инициализация, 2-45
    - Редактирование, 10-4
    - Редактирование для тэга, 2-22
    - Редактирование тем, 10-9
    - Сервер ввода-вывода InSQL, 10-18
    - Синхронизация метки времени, 10-12
    - Удаление, 10-8
  - Сетевая библиотека, 11-6, 12-11, 12-13
    - описание, 12-6
    - Стандартная, 12-8
  - Сетевое взаимодействие
    - IndustrialSQL Server, 11-2
  - Сетевое соединение
    - IndustrialSQL Server, 11-7
  - Сетевые протколы
    - IPX/SPX, 12-11
  - Сетевые протоколы
    - IPX/SPX, 11-7
    - Multi-protocol, 11-7, 12-13
    - named pipes, 11-7, 12-9
    - TCP/IP, 11-7, 12-15
  - Сеть процесса, 1-7
  - Сжатие, 4-12
  - Сжатые данные, 4-12
  - Символы ANSI, 12-8
  - Символы OEM, 12-8
  - Синхронизация меток времени, 10-12, 10-16
  - Система
    - Запуск, 5-7
    - Мониторинг, 8-2
  - Система управления событиями, 5-4
    - Детекторы событий, 7-4
    - Компоненты, 7-9
    - Описание, 1-12, 7-2

Переменные, 7-10  
 Таблицы сводок, 7-12  
 Таблицы хранения событий, 7-11  
 Система хранения  
   Остановка, 5-10  
 Системное "здоровье", 5-4  
 Системные объекты, 2-8  
 Системные ошибки, 5-5  
 Системные переменные, 8-3  
 Системные процессы, 4-16, 5-8  
 Системные тэги, 3-10  
 Системный администратор  
   Пароль, 13-19  
 Системный драйвер, 5-5  
 Системный драйвер IndustrialSQL, 1-14  
 Системный драйвер мониторинга, 8-3  
   Запуск, 8-3  
 Системный сервер ввода-вывода, 2-43, 10-2  
 Скорость обработки данных, 5-2  
 Скорость сканирования, A-11  
 Скорость сохранения, 3-12, 4-2  
 Словарь тэгов  
   Импортирование, 3-2  
 Службы, 4-19, 5-17, A-7, A-13  
 Службы открытых данных (ODS), 6-2  
 Снимок  
   Настройка, 7-28  
 Событие  
   Определить тэг, 2-42  
 События, 1-6, A-13  
   Архивная запись, 7-9  
   Настройка, 7-13  
   Описание, 7-2  
 Соединения клиента  
   Настройка, 12-8  
   Настройка общих параметров, 12-8  
 Соединения клиентов, 11-6  
   Настройка, 12-2  
   Устранение проблем, 12-16  
 Создание сводки  
   Настройка, 7-34  
 Сообщения  
   Просмотр, 5-14  
 Сообщения об ошибках, 8-10, A-5  
 Сообщения состояния  
   Просмотр, 5-14  
 Сохранение данных  
   Скорость записи в активный образ, 4-3  
 Статистика сбора данных, 5-13  
 Строка соединения, 12-2, 12-9, 12-12, 12-13, 12-14  
 Строка состояния  
   InSQL Control, 5-4, 5-5  
   Показать/скрыть, 2-12  
 Счетчик пакетов, 5-8  
   Отслеживание, 5-11  
 Счетчики, 8-7  
**T**  
 Таблица *StorageLocation*, 4-8  
 Таблица *SummaryData*, 2-28, 7-12  
 Таблицы расширения, 1-9, 1-11  
   Хранение, 4-7  
 Текстовое хранение, 5-4  
 Текстовые значения

Период хранения, 2-14  
 Текстовый тэг  
   Добавить, 2-41  
   Настройка, 2-39  
   Повторная инициализация, 2-44  
   Редактировать общие свойства, 2-39  
 Тема, 10-2  
   Добавление для сервера ввода-вывода, 10-10  
   Редактирование для сервера ввода-вывода, 10-9  
   Редактировать для тэга, 2-22  
   Удаление, 10-11  
 Темы  
   Импортирование, 3-11  
   Просмотр, 5-14  
 Типичные ошибки, 14-2  
 Триггеры, 1-9  
 Тэг  
   Добавить аналоговый тэг, 2-29  
   Добавить дискретный тэг, 2-35  
   Добавить текстовый тэг, 2-41  
   Конфигурация сбора данных, 2-21  
 Тэг события  
   Настройка общей информации, 7-13  
   Описание, 2-42, 7-3  
   Повторная инициализация, 2-45  
   Создание, 7-16  
   Удаление, 7-42  
 Тэги  
   Импортирование, 3-3  
   Настройка, 2-21  
   Фильтрация в броузере, 2-9

## У

Удаленные таблицы, 1-9, 1-11  
   Справка, 6-5  
   Хранение, 4-7  
 Узел InTouch  
   Импортирование, 3-3  
 Узлы хранения, 4-5, A-2, A-3  
 Унифицирующий символ, 3-9  
 Устройства хранения, 4-19  
 Утилита SQL Server Client Network, 12-3  
 Утилита резервного копирования, 1-15  
 Учетная запись, 5-18  
 Учетная запись пользователя, 5-17  
 Учетные записи. *См. также* Идентификаторы пользователей  
   Стандартные в IndustrialSQL, 13-6  
   Управление, 11-16

## Ф

Файл TAGNAME.X, 2-2  
 Файл базы данных, 4-7  
   Просмотр, 4-5  
 Файл журнала, 4-5, 14-3, 14-6  
 Фатальные ошибки, 14-2  
 Фильтрация, 2-9

## Х

Хранение данных, A-5, A-10  
   Архивные данные, 4-7

Данные конфигурации, 4-5  
Запуск, 4-16  
Определение дискового пространства, 4-8  
Оптимизация, 4-23  
Перевод часов на зимнее/летнее время, 4-20  
Размер диска, 4-13

Хранилище данных, 6-4

## Ц

Целое значение, 2-22  
Циклическая выборка, 1-11, 6-2  
Циклическая область хранения, 4-8, 4-15, А-3  
    Описание, 4-11  
Циклическое сохранение, 3-12  
Циклическое хранение, 2-26, 2-27  
    Изменение метки времени, 10-15  
    Скорость, 2-27  
Циклическое хранение данных  
    Описание, 4-2

## Ч

Частота, 2-16

## Ш

Шаблоны SQL, 7-20  
Широковещательные пакеты, 12-11  
Шлюз, 10-20  
Шлюзовые серверы, 6-3

## Э

Электронное сообщение  
    Настройка, 7-29  
Элемент, 10-2  
Элементы ActiveX, 3-14