



InduSoft Web Studio™ v7.0



Содержание

ВВЕДЕНИЕ.....	3
Описание продукта.....	4
Технические характеристики продукта.....	5
Условные обозначения, используемые в этой документации.....	8
УСТАНОВКА.....	10
Системные требования.....	11
Установка программного обеспечения.....	14
Запуск программы.....	19
Удаление программного обеспечения.....	20
СРЕДА РАЗРАБОТКИ.....	21
Заголовок окна (Title Bar).....	22
Строка состояния (Status Bar).....	23
Кнопка приложения (Application button).....	24
Панель инструментов быстрого доступа (Quick Access Toolbar).....	25
Лента (Ribbon).....	27
Вкладка Home.....	27
Вкладка View.....	28
Вкладка Insert.....	28
Вкладка Project.....	29
Вкладка Graphics.....	29
Вкладка Format.....	30
Вкладка Help.....	31
Проводник проекта (Project Explorer).....	32
Секция Global.....	32
Секция Graphics.....	34
Секция Tasks.....	35
Секция Comm.....	36
Редактор экранных форм (Screen/Worksheet Editor).....	38
ИНФОРМАЦИЯ О ТЕГАХ И БАЗЕ ДАННЫХ ПРОЕКТА.....	39
Правила написания имен тегов.....	41
Типы данных тегов.....	42
Массивы тегов.....	44
Косвенные теги (indirect tags).....	49
Список свойств тегов.....	51
Сброс базы данных тегов.....	61

УЧЕБНОЕ РУКОВОДСТВО: СОЗДАНИЕ ДЕМОНСТРАЦИОННОГО ПРОЕКТА.....	62
Создание нового проекта.....	63
Установка стартового экрана.....	65
Создание тегов.....	67
Создание стартового экрана.....	69
Создание заголовка стартового экрана.....	70
Рисование кнопки для открытия другого экрана.....	72
Сохранение и закрытие стартового экрана.....	74
Создание экрана мнемосхемы.....	75
Рисование заголовка экрана мнемосхемы.....	75
Отображение текущего времени и даты.....	76
Размещение значка "выход".....	77
Тестирование проекта.....	79
Размещение анимированной емкости.....	79
Размещение слайдера уровня.....	82
Рисование селектора емкостей.....	83
Тестирование проекта.....	84
Конфигурирование драйвера связи.....	86
Отслеживание обмена данными с устройством ввода/вывода в режиме исполнения.....	90
Загрузка проекта на устройство Windows Embedded.....	91
Развертывание проекта как web-приложения.....	94

Введение

InduSoft Web Studio (далее IWS) является мощным, интегрированный инструментом, который, используя ключевые возможности операционных систем Microsoft, позволяет создавать полнофункциональные приложения SCADA (Supervisory Control and Data Acquisition) и приложения HMI (Human-Machine Interface) для решения задач автоматизации в различных сферах деятельности.

Это Руководство по быстрому освоению *InduSoft Web Studio* предназначено для тех пользователей, которые впервые используют IWS. Оно поможет вам быстро ознакомиться с основными возможностями IWS и приступить к работе.

Описание продукта

Проекты InduSoft Web Studio работают на компьютерах, подключенных к программируемым логическим контроллерам (ПЛК), устройствам удаленного ввода/вывода (I/O) и другому оборудованию для сбора данных и управления.

Эти проекты содержат анимированные экраны операторского интерфейса, конфигурируемые драйверы ПЛК и других I/O устройств, базу данных тегов, дополнительные модули, такие как мониторы аварий (alarm monitors), логику (logic), тренды (trend charts), рецепты (recipes), планировщики (schedulers) и подсистему безопасности. Во время работы проекты IWS взаимодействуют с системами ввода/вывода и другими приложениями Windows с использованием следующих технологий и протоколов:

- ODBC (*Open Database Connectivity*)
- DDE (*Dynamic Data Exchange*)
- NetDDE (*NetworkDynamic Data Exchange*)
- OPC (*OLE for Process Control*)
- TCP/IP (*Transmission Control Protocol/Internet Protocol*)

После создания проекта вы можете запускать его как в среде разработки (на рабочей станции разработчика), так и в среде исполнения (рабочее место оператора). Загружать проект в среду исполнения (InduSoft Web Studio или CEView runtime) можно с помощью последовательного или TCP/IP соединения. Процессы, исполняемые в рамках проекта InduSoft Web Studio на рабочей станции оператора или на сервере SCADA, могут опрашивать подключенные устройства в соответствии с конфигурацией проекта, реагировать на полученные данные, отображать их на экране, сохранять и передавать их в другие системы.

Программный продукт InduSoft Web Studio состоит из двух частей:

- Среда разработки (development system) – программное обеспечение, запускаемое на настольном, переносном или промышленном компьютере под управлением (поддерживаемой в настоящий момент) настольной или серверной операционной системы Microsoft Windows.
- Среда исполнения (runtime system) – программное обеспечение, запускаемое на операторской рабочей станции под управлением текущей настольной, серверной или встраиваемой (embedded) операционной системы Microsoft Windows.



Примечание: Среда исполнения для Windows Embedded (CEView) часто может быть предустановлена на HMI-устройстве.

Если необходимо, вы можете обновить версию CEView, загрузив обновление в HMI-устройство.

Технические характеристики продукта

InduSoft Web Studio состоит из следующих основных компонентов:

- Интегрированная многооконная [среда разработки](#) с панелями инструментов, диалоговыми окнами и меню:
 - Всплывающие (pop-up) меню, активирующиеся по нажатию правой кнопки мыши в любой области среды разработки (меняются в зависимости от контекста)
 - Настраиваемые плавающие [панели инструментов](#)
 - Задачи, объекты и элементы управления (controls), доступные в древовидном меню [проводника проекта](#) (explorer)
- Многофункциональные объекты и анимация (с возможностью менять свойства объектов, выполнять команды, присваивать значения тега во время исполнения):
 - Конфигурируемые объекты, такие как кнопки, геометрические фигуры, текст
 - [Анимированные объекты](#), динамически изменяющие свои пропорции, цвет, размер и положение на экране, прозрачность и видимость, гиперссылки, текстовые блоки
 - Отображение [списка тревог](#) (текущего и исторического)
 - Отображение [трендов](#) (текущих и исторических)
 - Инструменты [выравнивания и сортировки](#) объектов
 - Средства создание и редактирования [фонового растрового изображения](#)
 - Средства импортирования графических объектов
 - Контейнеры [ActiveX](#) объектов
- Средства удаленного on-line конфигурирование и управление
- Средства обеспечения совместимости с архитектурой Microsoft DNA, с полной поддержкой OPC и XML
- Средства поддержки Web-интерфейса, через который "тонкие клиенты" получают доступ к экранам проекта в сетях Internet/Intranet по протоколам TCP/IP

- **Библиотека символов**, включающая более 100 готовых объектов, таких как рамки, кнопки, измерители, ползунковые регуляторы, переключатели, алфавитно-цифровые индикаторы, светодиодные индикаторы, трубопроводная арматура, насосы, двигатели, клапаны, КИП
- Инструменты отладки:
 - **Анализатор базы данных** (Database Spy window) для просмотра/изменения значений тегов и выполнения запросов
 - **Журнал трассировки** (LogWin module) для записи транзакций OPC, DDE и TCP/IP, активности модулей, трассировки тегов и т.п.
 - **Перекрестные ссылки** (Cross-referencing) для поиска тегов во всем проекте
 - **Онлайновая системная и сетевая диагностика**
- Мощная и гибкая база данных тегов с поддержкой Boolean, Integer, Real и String типов данных тегов, массивов тегов, классов, косвенных указателей на теги
- Открытая архитектура, наличие интерфейса (API) для обмена тегами с внешними приложениями
- **Редактор переводов**, который позволяет переводить проект на различные языки и переключаться между ними во время работы в среде исполнения
- **TCP/IP** клиент и серверные модули для обмена значениями тегов и конфигурирования систем с резервированием
- Более 200 **коммуникационных драйверов** для различных устройств (таких как ПЛК) от разных производителей, таких как Allen-Bradley, Siemens, GE-Fanuc, так же для стандартных протоколов обмена, таких как MODBUS RTU/ASCII, DeviceNet, Profibus, Interbus, и т.п.
- Полная интеграция с PC-совместимыми пакетами управления (импорт тегов), такими как ISaGRAF, SteepleChase, Think&Do, OpenControl, FP Control и ASAP
- OPC сервер и OPC клиент с интегрированным **OPC браузером**
- **Система безопасности** на основе защиты паролями экранов и объектов (256 уровней)
- Логические выражения и **язык сценариев** с более чем 200 функций
- **Рецепты** и **генератор отчетов** (в формате ASCII, UNICODE и RTF)
- **Планировщик событий**, срабатывающий по дате, времени или условию перехода (временное разрешение 100мс.)

- Многоуровневый проект, основанный на модульных рабочих таблицах (worksheets) и экранах, которые могут быть объединены с другими проектами
- Подсистема поддержки [коммутируемых соединений](#) (Dial-Up) для создания, контроля и завершения соединений с RAS-серверами
- Встроенные функции для отправки [электронной почты](#) (e-mail) из IWS (или CEView)
- Документирование проекта в реальном времени
- [Конвертер экранного разрешения](#)



Примечание: IWS предоставляет различные варианты продукта для каждого уровня ответственности проекта. Однако IWS не поддерживает некоторые функции в определенных вариантах продукта. Для получения дополнительной информации смотрите [О целевых платформах и типах продукта](#).

Условные обозначения, используемые в этой документации

Для облегчения восприятия этот документ использует стандартизированное форматирование.

Оформление текста

Этот документ использует специальное форматирование текста, чтобы помочь вам быстрее ориентироваться в изложенной информации:

- Заголовки, метки, новые понятия, сообщения отображаются в виде наклонного шрифта (например, Свойства объекта (*Object Properties*)).
- Имена файлов, экранный текст, и текст, который следует вводить, обозначаются с использованием шрифта `monospace` (например, `D:\Setup.exe`).
- Кнопки, пункты меню, клавиши на клавиатуре обозначаются с использованием жирного шрифта (например, меню **File**).

Также в этом документе выделяется текст в Советах, Примечаниях и Предостережениях, в блоках, окруженных рамкой:

- Советы сообщают полезную информацию, которая экономит время разработки и увеличивает производительность проекта.
- Примечания предоставляют дополнительную информацию, которая может облегчить понимание окружающего текста.
- Предостережения - информация для предотвращения ошибок, которые могут вызвать проблемы в работающем проекте и привести к сбоям.

Обозначения работы с мышью и действий выбора

Поскольку большинство ПК, используемых для создания проекта, работает под управлением Microsoft Windows с манипулятором типа мышь, этот документ предполагает, что вы также используете мышь. Обычно у большинства пользователей мышь сконфигурирована для использования правой рукой, так что левая кнопка мыши является первой, а правая кнопка второй.

Этот документ использует следующие обозначения при работе с мышью:

- Щелчок (**Click**) и Выбор (**Select**) обозначают однократное нажатие на элемент (пункт) левой кнопкой мыши. В общем, вы нажимаете (щелкаете) кнопки и выбираете пункты меню или списка.
- Двойной щелчок (**Double-click**) обозначает быстрое двойное нажатие по элементу левой кнопкой мыши.

- Правый щелчок (**Right-click**) обозначает быстрое одинарное нажатие по элементу правой кнопкой мыши.
- Выбор (**Select**) обозначает, что вы должны использовать указатель мыши для выделения или обозначения элементов (пунктов) на компьютерном экране. Выбор элементов при помощи сенсорного экрана (touchscreen) обычно аналогичен выбору с помощью мыши, за исключением того, что вы используете нажатие пальца для указания на объект или область. Чтобы выбрать элементы или пункты с клавиатуры, обычно используется клавиша Tab для обхода опций, Enter для открытия пунктов меню и Alt совместно с буквенной клавишей для выбора объекта, который имеет в подписи соответствующую подчеркнутую букву.
- Перетаскивание (**Drag**) обозначает надавливание на соответствующую кнопку мыши и перемещение указателя перед отпусканием кнопки. В данном случае выбранный для перетаскивания объект перемещается вместе с курсором мыши.

Обозначения оконного интерфейса

Этот документ использует следующие обозначения для оконного интерфейса:


- Диалоги или диалоговые окна (**dialogs**) – окна, которые позволяют осуществлять настройки и вводить информацию.
- Текстовые поля (**Text boxes**) - области диалогового окна, в которые можно вводить текст.
- Переключатели (**Radio buttons**) - белые круглые элементы с черной точкой в центре, обозначающей выбор данного элемента. Наличие точки в центре обозначает включение (выбор). Отсутствие точки обозначает выключение (отсутствие выбора).
- Флаги (**Check boxes**) - белые квадратные элементы, в которых флаг появляется (Titlebar) или исчезает, когда вы щелкаете по ним курсором. Наличие флага в центре обозначает включение (выбор). Отсутствие флага обозначает выключение (отсутствие выбора).
- Кнопки (**Buttons**) – значки, обозначающие элементы управления кнопки.
- Списки (**Lists**) – области окна, содержащие две и более опций, которые можно выбрать.
- Поля со списком (**Combo boxes**) - раскрывающиеся списки с раскрывающей кнопкой.
- Стыкуемые окна (**Dockable windows**) - окна, которые можно отделять от края интерфейса и стыковать с ним.

Установка

В этом разделе содержатся инструкции по установке, запуску и удалению InduSoft Web Studio, EmbeddedView и CEView.

Системные требования

Минимальные системные требования для установки и запуска InduSoft Web Studio.

-  **Примечание:** Приведенные ниже требования базируются на типовых проектах. В зависимости от специфики вашего проекта требования могут варьироваться:
- Windows Embedded-совместимые устройства представлены системами с различными процессорами и функциями, от смартфонов до промышленных панелей. Проконсультируйтесь с вашим поставщиком для определения требований к аппаратным ресурсам для запуска проекта на этих устройствах.
 - Некоторые из пунктов заявлены в качестве опциональных, могут быть обязательными для вашего проекта. Например, если вам требуется обмен данными с ПЛК через последовательный интерфейс, компьютер, возможно, должен иметь последовательный порт.

Станция разработчика / Сервер проекта / Клиент проекта

Для установки и запуска полного пакета InduSoft Web Studio вам необходимо:

- Windows-совместимый компьютер или Windows Embedded-совместимое устройство со стандартной клавиатурой, мышью и SVGA-экраном.
- Операционная система Windows или Windows Embedded, которая в данный момент поддерживается Microsoft, в настоящее время это:
 - Microsoft Windows XP Service Pack 3 или более поздняя
 - Microsoft Windows Vista Service Pack 1 или более поздняя
 - Microsoft Windows 7, все версии
 - Microsoft Windows Server 2003 Service Pack 2 или более поздняя
 - Microsoft Windows Server 2008, все версии
 - Microsoft Windows XP Embedded Service Pack 3
 - Microsoft Windows Embedded Standard 7 (2009)

-  **Подсказка:** Мы рекомендуем "Professional" и "Ultimate" издания вместо "Home" and "Media Center", потому что они включают

Internet Information Services (IIS). Вам может понадобиться IIS, если вы хотите "развернуть" ваш IWS проект как web-приложение.

- Microsoft Internet Explorer 6.0 или более поздний
- Минимально 500МВ свободного дискового пространства
- Ethernet-адаптер или беспроводная сеть
- CD-ROM привод (опционально, для установки приложений, которые также могут быть загружены с нашего web-сайта)
- USB-порт (опционально, используется для аппаратного ключа лицензий)
- Последовательные COM порты и адаптеры (опционально, для прямого соединения с ПЛК и другими устройствами)

Любой компьютер, на котором установлен полный пакет InduSoft Web Studio, может быть сервером проекта и/или клиентом проекта. К ним относятся устройства Windows XP Embedded и Windows Embedded Standard, но в большинстве случаев, если вы не планируете вести разработку, может оказаться более практичным установить EmbeddedView на этих устройствах, так как последний занимает меньше места и может обслуживаться удаленно.

Вы не можете установить полный пакет InduSoft Web Studio на устройства с Windows Embedded Compact.

Сервер проекта – Встраиваемый (**Embedded**)

Для запуска сервера проекта, который является хостом среды исполнения (runtime) проекта, при использовании EmbeddedView или CEView, у вас должно быть в наличии:

- Windows Embedded-совместимое устройство
- Операционная система Windows Embedded, которая в данный момент поддерживается Microsoft, в настоящее время это:
 - Microsoft Windows XP Embedded Service Pack 3
 - Microsoft Windows Embedded Standard 7 (2009)
 - Microsoft Windows Embedded Compact (ранее известная как Windows CE или Windows Mobile), версия 5.0 или более поздняя
- Минимально 500МВ свободного дискового пространства
- Ethernet-адаптер или беспроводная сеть
- USB-порт (опционально, используется для аппаратного ключа лицензий)
- Последовательные COM порты и адаптеры (опционально, для прямого соединения с ПЛК и другими устройствами)

Клиент проекта – Встраиваемый (**Embedded**)

Для работы клиентской части с использованием EmbeddedView или CEView у вас должно быть в наличии:

- Windows Embedded-совместимое устройство с мышью или сенсорным экраном
- Операционная система Windows Embedded, которая в данный момент поддерживается Microsoft, в настоящее время это:
 - Microsoft Windows XP Embedded Service Pack 3
 - Microsoft Windows Embedded Standard 7 (2009)
 - Microsoft Windows Embedded Compact (ранее известная как Windows CE или Windows Mobile), версия 5.0 или более поздняя
- Ethernet-адаптер или беспроводная сеть

Клиент проекта – Тонкий клиент (**Thin**)

Для работы клиента проекта с использованием Secure Viewer либо браузерного тонкого клиента у вас должно быть в наличии:

- Windows-совместимый компьютер или Windows Embedded-совместимое устройство с мышью либо сенсорным экраном
- Операционная система Windows для рабочих станций, серверов или встраиваемых устройств, которая в данный момент поддерживается Microsoft, в настоящее время это:
 - Microsoft Windows XP Service Pack 3 или более поздняя
 - Microsoft Windows Vista Service Pack 1 или более поздняя
 - Microsoft Windows 7, все версии
 - Microsoft Windows Server 2003 Service Pack 2 или более поздняя
 - Microsoft Windows Server 2008, все версии
 - Microsoft Windows XP Embedded Service Pack 3
 - Microsoft Windows Embedded Standard 7 (2009)
 - Microsoft Windows Embedded Compact (ранее известная как Windows CE или Windows Mobile), версия 5.0 или позднее
- Microsoft Internet Explorer 6.0 или более поздний
- Ethernet-адаптер или беспроводная сеть

Установка программного обеспечения

Среда разработки InduSoft Web Studio предоставляет возможность разработки проектов IWS для всех типов поддерживаемых целевых платформ. Среда разработки может быть установлена на ПК под управлением операционной системы Microsoft Windows XP, Windows Vista или Windows 7. Для получения дополнительной информации смотрите раздел [Системные требования](#).

Среду разработки можно либо загрузить с web-сайта компании, либо установить с диска InduSoft Web Studio installation CD. Установленную на ПК среду разработки можно использовать для загрузки на целевую платформу среды исполнения CEView через последовательный порт или TCP/IP соединение.

Программа установки IWS создает необходимые директории, копирует файлы на жесткий диск и создает ярлык InduSoft Web Studio на рабочем столе.



Примечание:

- Вы должны иметь права администратора на вашем ПК для установки и удаления среды разработки.
- Вы должны удалить старую версию среды разработки (или переместить ее в другую директорию), прежде чем устанавливать новую версию. Кроме того, вы не можете установить ту же самую версию среды разработки в две различные директории на данном ПК.

Инструкция по установке InduSoft Web Studio и CEView приведена в следующих двух разделах.

Установка среды разработки на компьютер

Для установки среды разработки IWS с CD диска:

1. Включите компьютер и убедитесь, что никаких других программ не запущено.
2. Вставьте установочный диск в CD-ROM привод.

Internet Explorer должен автоматически запуститься и показать экран приветствия CD. Если этого не произошло, например, если у вас отключена опция автозапуска для CD-ROM, вы можете вручную открыть экран приветствия с помощью Windows Explorer, открыв файл D:\InduSoft.htm.

3. На экране приветствия выберите продукт, который вы хотите установить.

Internet Explorer спросит, хотите ли вы запустить или сохранить файл установщика.

4. Щелкните **Run**.

Мастер установки продукта начнет свою работу.

5. Следуйте инструкциям мастера в процессе установки.

6. Когда установка завершится, выберите "Да, перезагрузить компьютер сейчас" и щелкните **OK**.

После перезагрузки ПК можете запустить среду разработки. Смотрите [Запуск программы](#) для соответствующих инструкций.



Примечание: Вместе со средой разработки устанавливается Microsoft .NET Framework и некоторые другие утилиты поддержки IWS. На вашем ПК уже может быть установлена более поздняя версия .NET Framework, нет причин для беспокойства, потому что разные версии .NET Framework не должны конфликтовать между собой. Вы можете посмотреть, какие версии установлены, открыв "Добавление или удаление программ" в панели управления (**Start > Control Panel > Add or Remove Programs**).

Для получения дополнительной информации о Microsoft .NET Framework смотрите [Database Appendix A: Using ODBC Databases](#).

Установка **CEView** на устройство **Windows Embedded**

CEView – среда исполнения проектов IWS на устройствах Windows Embedded. CEView должна быть установлена на вашем устройстве прежде, чем вы начнете загружать на него проект.

Где искать файлы?

Для разных платформ Windows Embedded устройств и каждой комбинации версии операционной системы и типа процессора существует своя сборка CEView. Все сборки располагаются в директории:

Redist

Сборка вашего специфического устройства расположена в директории:


Redist\version\processor

...где:

- **version** - версия операционной системы устройства, на которое устанавливается CEView:
 - \WinCE 5.0 каталог содержит файлы для Windows CE and Windows Mobile 5.0 или более поздних версий
 - \WinEmbedded каталог содержит файлы для Windows XP Embedded и Windows Embedded 7.
- **processor** - процессор, используемый на вашем Windows CE устройстве. Мы предлагаем среду исполнения CEView для каждого процессора, который в данный момент поддерживается ОС Windows CE (например, Pocket2003-ArmV4, ArmV4i, x86). Для получения дополнительной информации обратитесь к документации от производителя вашего устройства.

Для установки файлов на устройство используйте инструмент [Remote Management](#) среды разработки.

Установка по **TCP/IP (Ethernet)**

 Подсказка: Мы рекомендуем по возможности использовать TCP/IP.

Для установки CEView на устройство Windows Embedded посредством TCP/IP (Ethernet) соединения:

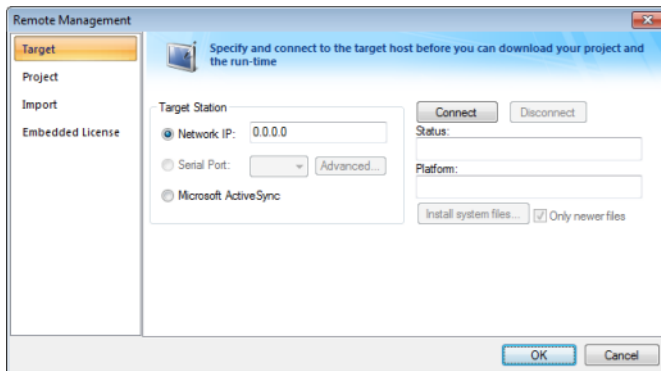
1. Удостоверьтесь, что Windows Embedded устройство подключено к вашей сети.
2. Включите устройство.

Диалоговое окно *Remote Agent* должно открыться автоматически. Если этого не произошло, следует вручную установить файл **CEServer.exe** на устройство. Файл располагается здесь:

Redist\version\processor\CEServer.exe

Есть различные способы установки файла, например, можно использовать Microsoft ActiveSync (для Windows 2000 или Windows XP) или Windows Mobile Device Center (для Windows Vista), чтобы связаться напрямую с устройством или отобразить устройство как совместную папку на вашем ПК. Для дополнительной информации о копировании и запуске файлов обратитесь к документации от производителя вашего устройства.

3. В диалоге *Remote Agent* щелкните по **Setup**, затем сконфигурируйте настройки сетевого соединения устройства. Обратите внимание на IP-адрес устройства.
4. Запустите среду разработки на ПК.
5. На вкладке **Home** ленты, в группе **Remote Management** щелкните **Connect**. Отобразится диалоговое окно *Remote Management*:



Диалоговое окно *Remote Management*

6. В групповом блоке *Target Station* выберите **Network IP** и затем введите IP-адрес устройства.
7. Щелкните **Connect**, чтобы установить соединение между средой разработки и устройством.
Если соединение произойдет успешно, спецификация устройства отобразится в текстовом поле **Platform**.
8. Щелкните **Install System Files** для загрузки файлов CEView на устройство.
9. Когда установка будет завершена, щелкните **Disconnect**.


Для дополнительной информации по загрузке и запуску конечного проекта на устройстве Windows Embedded смотрите [Remote Management](#).

Установка через **Microsoft ActiveSync**

Для установки CEView на устройство Windows Embedded через Microsoft ActiveSync:

1. Удостоверьтесь, что Microsoft ActiveSync (для Windows 2000 и Windows XP) или Windows Mobile Device Center (для Windows Vista) установлены на вашем ПК.

2. Включите устройство Windows Embedded и подсоедините его к вашему ПК. Большинство устройств имеют возможность подключаться через USB.
3. Запустите среду разработки на ПК.
4. На вкладке **Home** ленты, в группе **Remote Management** щелкните **Connect**. Отобразится диалоговое окно *Remote Management*.
5. В групповом блоке *Target System* выберите **Microsoft ActiveSync**.
6. Щелкните **Connect** для установления соединения между средой разработки и устройством.
Если соединение произойдет успешно, спецификация устройства отобразится в текстовом поле **Platform**.
7. Щелкните **Install System Files** для загрузки файлов CEView на устройство.
8. Когда установка будет завершена, щелкните **Disconnect**.

 **Примечание:** В некоторых случаях инструмент Remote Management не в состоянии соединиться посредством Microsoft ActiveSync с устройством Windows CE 6.0 или более поздней версии. Из-за проблемы в конфигурации, используемой по умолчанию в ОС Windows CE 6.0. Проблему можно решить с помощью небольшой утилиты, которая включена в IWS. Утилита располагается в:

Redist\ActiveSyncUnlock.exe

Скопируйте этот файл на устройство, используя автономную версию Microsoft ActiveSync, и затем выполните файл на устройстве. Не имеет значения, где на устройстве будет расположен этот файл. (Для дополнительной информации о копировании и запуске файлов обратитесь к документации от производителя вашего устройства.) По завершении попробуйте еще раз использовать инструмент Remote Management, чтобы подключиться к устройству.

Для дополнительной информации о загрузке и запуске вашего проекта на устройстве Windows Embedded смотрите раздел [Remote Management](#).

Запуск программы

Для запуска InduSoft Web Studio:

- Дважды щелкните значок InduSoft Web Studio v7.0+SP1 на рабочем столе; или
- Выберите **Start > All Programs > InduSoft Web Studio v7.0 > InduSoft Web Studio v7.0**.




Подсказка: Вы можете запустить среду разработки IWS при любом экранном разрешении ОС. Однако мы рекомендуем использовать разрешение 800x600 (или выше), и использовать более 256 цветов. Экранное разрешение проекта не зависит от экранного разрешения ОС.

Удаление программного обеспечения



Внимание: Прежде чем запустить процедуру удаления, убедитесь, что вы сохранили отдельно программные файлы, которые могут понадобиться позже. Также удостоверьтесь, пожалуйста, что вы имеете дистрибутив с текущей или более новой версией IWS, чтобы переустановить программное обеспечение в случае необходимости.

Если вам необходимо удалить IWS из системы, следуйте следующим инструкциям:

1. В панели задач Windows выберите **Start > Settings > Control Panel** для открытия Панели управления.
2. Дважды щелкните по значку **Add/Remove Programs** в окне Панели управления.
3. Когда отобразится диалог *Add/Remove Programs Properties*, выберите InduSoft Web Studio из списка и щелкните **Add/Remove**.
4. Когда появится диалог *Confirm File Deletion*, щелкните **Yes**.
Отобразятся диалоги *Uninstall Shield Wizard* и *Remove Programs from Your Computer*.
5. Когда отобразится сообщение об успешном удалении *Uninstall successfully completed*, и кнопка **OK** станет активной, щелкните **OK**.
Проверьте, что InduSoft Web Studio больше не присутствует в списке диалога *Add/Remove Programs Properties*.
6. Щелкните кнопку **Cancel** или закрывающую кнопку в углу окна () для закрытия диалога *Add/Remove Programs Properties*, затем закройте окно Панели управления.
7. Откройте *Windows Explorer* и перейдите к программной директории IWS.
8. Проверьте, что все IWS файлы и директории были удалены. (Вы можете вручную удалить любые оставшиеся файлы или директории.)

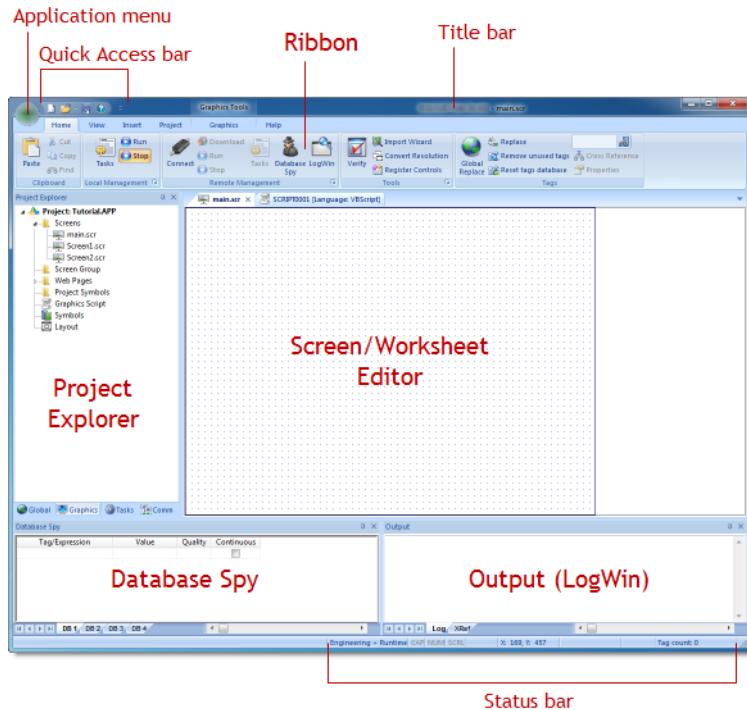


Примечание: Инструмент удаления не может удалить файлы, которые вы создали или модифицировали в директории для размещения ваших проектов (projects folder) IWS.

Вы должны обладать правами администратора, чтобы устанавливать или удалять InduSoft Web Studio.

Среда разработки

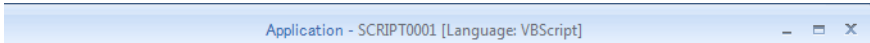
В InduSoft Web Studio присутствует современный "Ribbon" (ленточный) Windows интерфейс для создания интегрированной и дружелюбной среды разработки.



Среда разработки *IWS*





Заголовок окна (Title Bar)


Заголовок окна расположен вдоль верхней части среды разработки. В нем отображается имя приложения (например, InduSoft Web Studio), за ним следует название активного экрана или рабочей таблицы (worksheet) (если таковые имеются).



Пример заголовка окна

В заголовке окна также присутствуют следующие кнопки (слева направо):

- Кнопка **Minimize** , щелчок минимизирует окно в панель задач.
- **Restore Down / Maximize**, щелчок ведет к переключению между двумя размерами окна:
 - Кнопка **Restore Down**  уменьшает окно до оригинального (по умолчанию) размера.
 - Кнопка **Maximize**  увеличивает окно до полного размера компьютерного экрана.
- Кнопка **Close** , щелчок сохраняет базу данных и затем закрывает среду разработки. Если вы изменяли какие-нибудь экраны или рабочие таблицы, приложение предложит вам сохранить изменения. Действие этой кнопки подобно щелчку по **Exit Application** в меню Application.

 **Примечание:** Закрытие среды разработки не закрывает ни project viewer, ни среду исполнения (runtime system), если они были запущены.

Строка состояния (Status Bar)

Строка состояния расположена вдоль нижней части среды разработки, в ней отображается информация об активном экране (если существует) и состоянии приложения.

Engineering + Runtime | CAP | NUM | SCRL | ID: 0 | X: 160, Y: 90 | W: 201, H: 181 | Tag count: 0

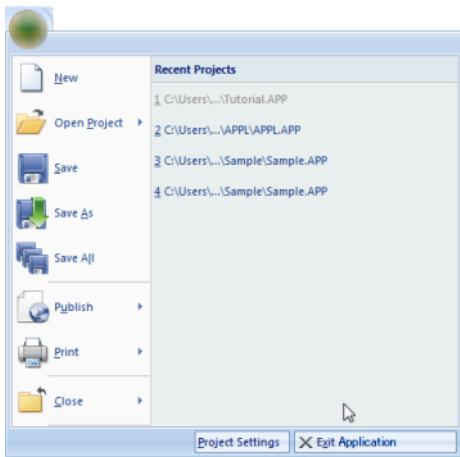
Пример строки состояния

Поля строки состояния (слева направо) описаны в следующей таблице:

Поле	Описание
Режим исполнения	Текущий режим исполнения приложения.
CAP	Показывает состояние клавиатуры Caps Lock . Включен (черный), выключен (серый).
NUM	Показывает состояние клавиатуры Num Lock . Включен (черный), выключен (серый).
SCRL	Показывает состояние клавиатуры Scroll Lock . Включен (черный), выключен (серый).
ID объекта	Идентификационный номер выбранного экранного объекта.
Координаты курсора	Координаты курсора на активном экране или рабочей таблице. Если выбран экран, координаты курсора мыши представлены как X,Y , где X – кол-во пикселей от левого края экрана, а Y – количество пикселей от верхней части экрана. Если выбрана рабочая таблица, то дается позиция текстового курсора: строка и столбец.
Размер объекта	Размер (в пикселях) выбранного на экране объекта, где W – ширина, H – высота.
Нет перетаскивания	Указывает на запрет перетаскивания (No DRAG) или разрешение (пусто) в активном экране.
Счетчик тегов	Общее кол-во тегов, задействованных в проекте.

Кнопка приложения (**Application button**)

Кнопка приложения открывает меню стандартных команд Windows-приложения, например, New, Open, Save, Print и Close.



Кнопка приложения открывает меню команд

Панель инструментов быстрого доступа (**Quick Access Toolbar**)

Панель инструментов быстрого доступа – настраиваемая панель инструментов, которая содержит набор команд, которые не зависят от ленточных вкладок (ribbon tab), отображаемых в данный момент.

Перемещение панели быстрого доступа

Панель быстрого доступа может располагаться в одном из двух мест:

- Верхний левый угол рядом с кнопкой приложения (положение по умолчанию);
- Ниже ленты, где она может распределиться по всей длине окна приложения.

Если вы не хотите, чтобы панель быстрого доступа отображалась в данном месте, вы можете переместить ее в другое:

1. Щелкните **Customize Quick Access Toolbar** .
2. Из списка щелкните **Show Below Ribbon** или **Show Above Ribbon**.

Добавление команд в панель быстрого доступа

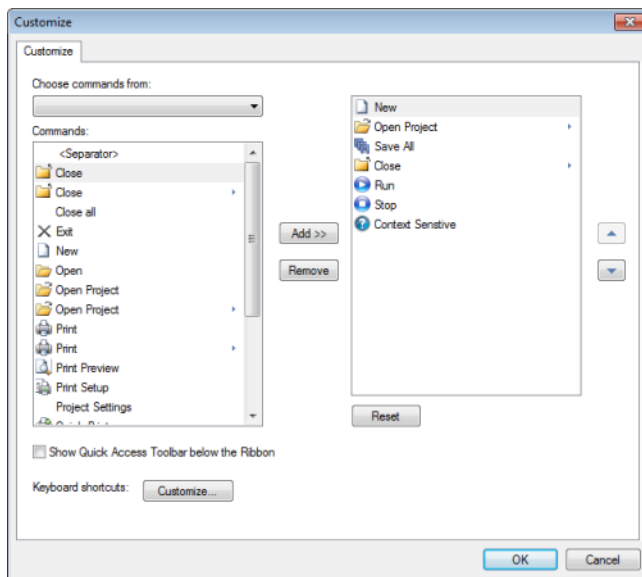
Вы можете добавить команду в панель быстрого доступа непосредственно от команды, отображаемой на ленте:

1. На ленте щелкните вкладку или группу с желаемой командой.
2. Щелкните правой кнопкой мыши по выбранной команде, и выберите и щелкните **Add to Quick Access Toolbar** в контекстном меню.

Кроме того, вы можете удалять и добавлять команды, а также сбрасывать панель в состояние по умолчанию, используя диалог Настройка (*Customize*):

1. Щелкните **Customize Quick Access Toolbar** .

2. В списке щелкните **More Commands**. Диалог *Customize* отобразится.



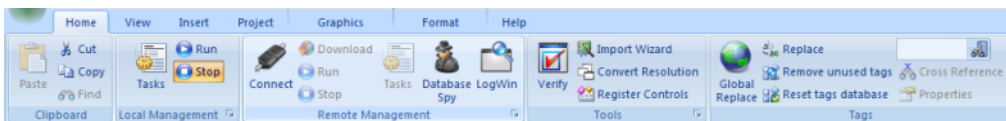
Диалог настройки панели быстрого доступа

3. В меню **Choose commands from** выберите желаемую вкладку. Команды соответствующей ленточной вкладки отобразятся в списке **Commands**.
4. В списке **Commands** выберите команду, которую вы хотите добавить в панель быстрого доступа.
5. Щелкните **Add**.

Только команды могут быть добавлены в панель быстрого доступа. Пункты большинства списков, такие как "отступ", "величина интервала", "индивидуальные стили", которые также присутствуют на ленте, не могут быть добавлены в панель быстрого доступа.

Лента (Ribbon)

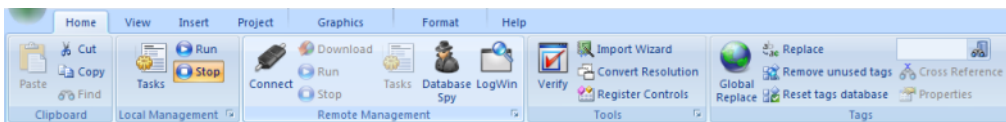
Новый ленточный интерфейс (далее Лента) комбинирует многочисленные меню и панели инструментов предыдущих версий IWS в единый, дружелюбный интерфейс. Почти все команды приложения теперь находятся на ленте, организованы во вкладки и группы в соответствии с общим принципом.



Ленточный интерфейс

Вкладка *Home*

Вкладка **Home** ленты используется для управления проектом в среде разработки.



Вкладка *Home*

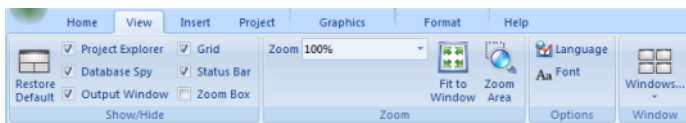
Инструменты организованы в следующие группы:

- Буфер обмена (**Clipboard**): **вырезать** (cut), **копировать** (copy), **вставить** (paste) и **найти** (find) элементы на экранах проекта и рабочих таблицах.
- Локальное управление (**Local Management**): **запустить** (run) и **остановить** (stop) проект на локальной рабочей станции (т.е. там, где установлена среда разработки), также тут осуществляется управление **выполняемыми задачами** (execution tasks).
- Удаленное управление (**Remote Management**): **подключиться** (connect) к удаленной рабочей станции (например, устройству Windows Embedded). Тут вы можете загружать проект, **запускать** (run), **останавливать** (stop) и **диагностировать проблемы** (troubleshoot).

- Инструменты (**tools**): различные инструменты для **проверки** (verify), **импорта тегов** (import tags) из других проектов, **конвертации разрешений экрана** и **регистрации ActiveX** и **.NET** компонентов.
- Теги (**Tags**): **управление тегами и свойствами тегов** в базе данных проекта.

Вкладка **View**

Вкладка **View** используется для настройки внешнего вида среды разработки.



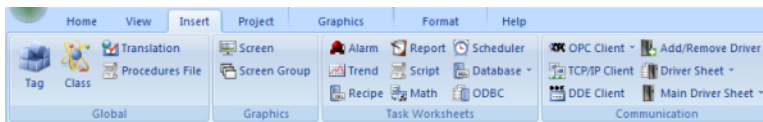
Вкладка **View**

Инструменты организованы в следующие группы:

- Показать/Скрыть (**Show/Hide**): показывает и скрывает различные части среды разработки, также восстанавливает значения по умолчанию макета.
- Масштабирование (**Zoom**): **масштабирование** в экранном редакторе.
- Опции (**Options**): изменяет **язык** (language) и **шрифт** (font), используемые в среде разработки.
- Окно (**Window**): **упорядочивает окна** в среде разработки.

Вкладка **Insert**

Вкладка **Insert** используется для добавления новых тегов, экранов, рабочих таблиц и других компонентов в проект.



Вкладка **Insert**

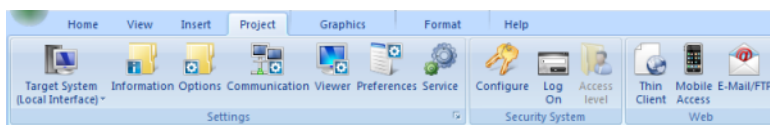
Инструменты организованы в следующие группы:

- Глобальные (**Global**): добавляет **теги** (tags), **классы** (classes), **переводы** (translations) и **процедуры** (procedures) в секцию **Global** проводника проекта (Project Explorer).

- **Графика (Graphics):** добавляет [экраны](#) (screens) и [группы экранов](#) в секцию [Graphics](#) проводника проекта.
- **Таблицы задач (Task Worksheets):** добавляет [таблицы задач](#) в секцию [Tasks](#) проводника проекта.
- **Связь (Communication):** добавляет [конфигурации серверов](#) и [коммуникационные таблицы](#) (server configurations и communication worksheets) в секцию [Comm](#) проводника проекта.

Вкладка **Project**

Вкладка **Project** используется для конфигурирования настроек проекта.



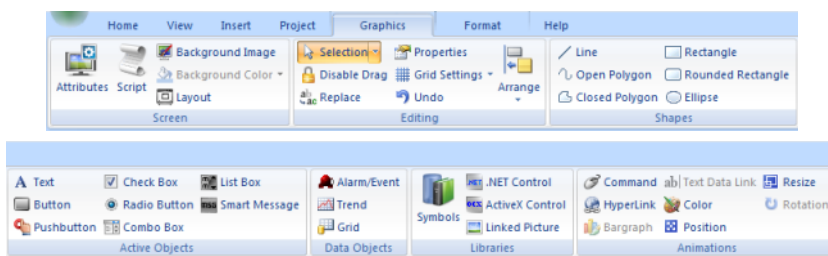
Вкладка **Project**

Инструменты организованы в следующие группы:


- **Настройки (Settings):** конфигурирует общие [настройки проекта](#) (project settings) и настраивает [запуск проекта в качестве службы Windows](#).
- **Система безопасности (Security System):** включает и конфигурирует [систему безопасности проекта](#).
- **Web:** конфигурирует подключения [тонких клиентов](#) и [мобильных устройств](#), а также исходящую [электронную почту](#) и [FTP](#).

Вкладка **Graphics**

Вкладка **Graphics** используется для рисования экранов проекта.



Вкладка **Graphics**

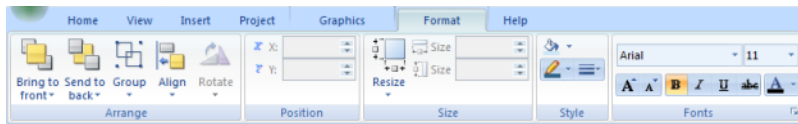
 **Примечание:** Эта вкладка доступна, только когда открыт экран проекта для редактирования.

Инструменты организованы в следующие группы:


- Экран (**Screen**): конфигурирует настройки экрана проекта, такие как [атрибуты](#) (attributes), [сценарии](#) (script), [фоновый цвет или изображение](#).
- Редактирование (**Editing**): [выбирает и редактирует объекты](#) на экране проекта.
- Фигуры (**Shapes**): рисует [статические линии и фигуры](#).
- Активные объекты (**Active Objects**): рисует [активные объекты](#), такие как кнопки и флажки.
- Объекты данных (**Data Objects**): рисует объекты, которые [отображают исторические данные](#), такие как alarms, events и trends.
- Библиотеки (**Libraries**): выбирает из библиотек готовые объекты, такие как [символы](#) (symbols), [.NET](#) и [ActiveX компоненты](#) и [внешние файлы изображений](#).
- Анимация (**Animations**): задействует [анимацию](#) для разных объектов на экране.

Вкладка **Format**

Вкладка **Format** используется для форматирования и задания размещения объектов на экране проекта.



Вкладка **Format**

 **Примечание:** Эта вкладка доступна, только когда вы выбираете один или несколько объектов на экране проекта.

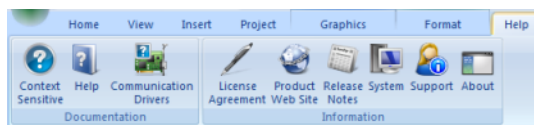
Инструменты организованы в следующие группы:

- Размещение (**Arrange**): размещает объекты на экране проекта, включая [перемещение на передний и задний планы](#), [группировка](#), [выравнивание](#) и [вращение](#).

- Позиция (**Position**): задает точные [координаты](#) объекта на экране проекта.
- Размер (**Size**): задает точный [размер](#) объекта на экране.
- Стил (**Style**): изменяет [заливку](#) и [цвет линий](#) объекта на экране.
- Шрифты (**Fonts**): изменяет [шрифт подписи](#) объекта на экране.

Вкладка **Help**

Вкладка **Help** предоставляет дополнительную информацию по использованию программного обеспечения.



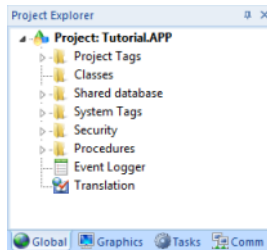
Вкладка **Help**

Инструменты организованы в следующие группы:

- Документация (**Documentation**): предоставляет доступ к документации для разработки, включая этот [справочный файл / технический справочник](#) и [документации по коммуникационным драйверам](#).
- Информация (**Information**): предоставляет доступ к другой информации об InduSoft Web Studio, включая [лицензионное соглашение](#), [web-сайт продукта](#), [примечания к релизу](#), доступ к детальной информации по системе ([system](#) и [support](#)), которая облегчает работу службы поддержки.

Проводник проекта (**Project Explorer**)

Проводник проекта организует работу со всеми экранами, рабочими таблицами и другими компонентами, представляет проект в виде расширяемого дерева.



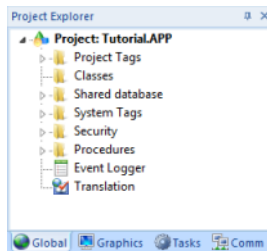
Проводник проекта

Щелкните один раз по значку "раскрыть" ▷ или двойным щелчком по папке для просмотра содержимого. Щелкните значок "схлопывание" ◀ для закрытия папки.

Если вы щелкните правой кнопкой мыши по любому компоненту в проводнике проекта, на экране появится контекстное меню с опциями для этого компонента.

Секция **Global**

Секция **Global** содержит базу данных тегов проекта, а также функции, которые применяются ко всему проекту, такие как безопасность и перевод интерфейса.



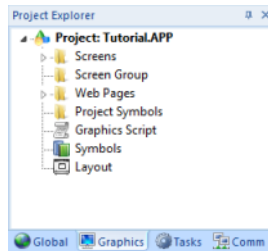
Секция **Global** проводника проекта

Папки секции **Global** описаны на следующих страницах:

- **Теги проекта (Project Tags)** содержит теги, которые вы создаете во время разработки проекта (такие как экранные теги или теги для чтения/записи данных из устройств ввода/вывода и подключенного оборудования).
- **Классы (Classes)** содержит составные теги, именуемые классами тегов, создаваемые для связывания наборов величин (а не одной величины) с объектами проекта.
- **Разделяемая база данных (Shared Database)** содержит теги, которые были созданы в PC-совместимых управляющих программах и затем импортированы в базу данных проекта.
Например, вы можете импортирования теги из SteepleChase в ваш проект таким образом, чтобы упростить дальнейшую настройку обмена данными вашего проекта с оборудованием, работающим под управлением программы, разработанной в SteepleChase.
- **Системные теги (System Tags)** содержит готовые теги с предопределенными функциями, которые могут быть использованы в проекте для разных задач (например, тег *Date* содержит текущую дату в формате строки).
Все системные теги доступны в проекте только для чтения, что означает, что вы не можете добавлять, редактировать или удалять их из базы данных.
- **Безопасность (Security)** содержит групповые и индивидуальные (пользовательские) учетные записи безопасности, сконфигурированные для данного проекта.
- **Процедуры (Procedures)** содержит VBScript функции и подпрограммы, которые можно вызывать в любом сценарии (script) в проекте.
- **Регистратор событий (Event Logger)** содержит настройки подсистемы журналирования событий.
- **Перевод (Translation)** содержит таблицу перевода, которая определяет, как пользовательский интерфейс проекта, разработанного на одном из языков, должен быть переведен при переключении системы исполнения на другой язык.

Секция **Graphics**

Секция **Graphics** проводника проекта содержит все экраны, группы экранов и символы (Symbols) проекта.



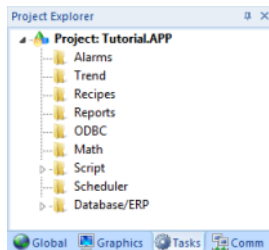
Вкладка **Graphics** проводника проекта

Папки секции *Graphics* описаны на следующих страницах:

- Экраны (**Screens**) содержит все [экранные формы](#), созданные в данном проекте.
- Группа экранов (**Screen Group**) содержит все [группы экранов](#) (индивидуальные экранные формы, объединенные в управляемые группы) данного проекта.
- **Web**-страницы (**Web Pages**) содержит все [Web-страницы](#) (т.е. экраны, сохраненные в HTML формате), созданные в проекте.
 - Мобильный доступ (**Mobile Access**) позволяет [конфигурировать мини-сайт](#), предназначенный для доступа с сотовых телефонов, смартфонов и других мобильных устройств.
- Символы проекта (**Project Symbols**) содержит все [пользовательские символы](#), которые могут быть группами изображений и/или текста. Вы можете создавать пользовательские символы и сохранять их в эту папку.
- Графические сценарии (**Graphics Script**) содержит [сценарии, которые запускаются, когда происходят определенные экранные события](#), например, при запуске тонкого клиента на удаленной рабочей станции.
- Символы (**Symbols**) содержит [библиотеку символов и графики](#), доступных для использования в проекте. Двойной щелчок по значку **Library** открывает Библиотеку символов.
- Макет (**Layout**) совместно отображает все открытые в редакторе экраны и позволяет [визуализировать совмещение экранов](#) для режима исполнения.

Секция **Tasks**

Секция **Tasks** проводника проекта содержит рабочие таблицы, которые обрабатываются в фоновых задачах во время выполнения проекта.



Секция **Tasks** проводника проекта

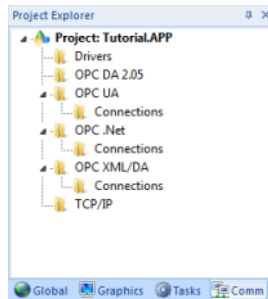
Папки секции **Tasks** описаны на следующих страницах:

- Тревоги (**Alarms**) содержит [таблицы тревог](#), которые используются для конфигурирования групп тревог и тегов, связанных с каждой группой тревог проекта. Задача тревог также используется для задания сообщений тревог.
- Тренды (**Trend**) содержит [таблицы трендов](#), которые используются для конфигурирования групп архивации (history groups) для сохранения данных трендов в проекте. С помощью этой секции можно указать теги, значения которых должны сохраняться на жёсткий диск, а также указать файлы, которые для этого должны быть использованы. Проект IWS может записывать данные трендов в двоичные файлы (*.hst) и одновременно отображать исторические и текущие значения тегов в окне графика трендов.
- Рецепты (**Recipes**) содержит [таблицы рецептов](#), которые используются для конфигурирования обмена данными между тегами проекта и файлами в формате ASCII или DBF на жёстком диске ПК.
- Отчеты (**Reports**) содержит [таблицы отчетов](#), которые используются для конфигурирования отчетов (текстового типа), которые могут быть распечатаны на принтере или сохранены в виде файла на жёстком диске ПК. Для повышения эффективности и упрощения создания отчетов в них могут быть включены также и системные параметры.
- **ODBC** содержит [ODBC таблицы](#), предназначенные для конфигурирования ODBC интерфейса в сетевом окружении с использованием стандартного окна конфигурирования ODBC ОС Windows. Конфигурирование ODBC необходимо для реализации обмена данными между проектом и любой базой данных, поддерживающей ODBC интерфейс.

- Вычисления (**Math**) содержит **таблицы вычислений**, которые используются для конфигурирования и написания дополнительных процедур для работы в различных задачах. В процессе работы проект выполняет вычисления в таблицах *Math* в фоновом режиме. Таблицы *Math* предназначены для задания произвольных арифметических или логических выражений, которые могут быть использованы в вашем проекте.
- Сценарии (**Script**) содержит **стартовый сценарий** (Startup Script) и другие **группы сценариев**.
- Планировщик (**Scheduler**) содержит **таблицы планировщика**, предназначенные для конфигурирования событий, могут запускаться по значению математического или логического выражения, по времени, дате или другому событию.
- Базы данных (**Database**)/ERP содержит **таблицы баз данных**, которые конфигурируют связь переменных (тегов) проекта со значениями внешних баз данных с использованием интерфейса ADO.NET (в качестве альтернативы ODBC).

Секция **Comm**

Секция **Comm** проводника проекта содержит рабочие таблицы, которые конфигурируют соединения с другими устройствами или программами с использованием доступных протоколов обмена.



Секция **Comm** проводника проекта

Папки секции **Comm** описаны на следующих страницах.

- Драйверы (**Drivers**) содержит **таблицы драйверов**, которые используются для конфигурирования интерфейса связи (интерфейсов) между проектом и удаленным оборудованием (таким как ПЛК).

Коммуникационный драйвер это **.DLL** файл, который содержит реализацию специфического протокола передачи данных для данного оборудования.

- **OPC DA 2.05** содержит [таблицы OPC](#), которые используются для конфигурирования связи между проектом (выступающим в роли OPC-клиента) и любым другим устройством, отвечающим классическому стандарту OPC-сервера.
- **OPC UA** содержит [таблицы OPC UA](#), которые используются для конфигурирования соединения с OPC-серверами по новому протоколу OPC Unified Architecture.
- **OPC .Net** содержит [таблицы OPC .Net](#), которые используются для конфигурирования соединения с OPC-серверами по новому протоколу OPC .NET 3.0.
- **OPC XML/DA** содержит [таблицы OPC XML/DA](#), которые используются для конфигурирования соединения с OPC-серверами по новому протоколу OPC XML-DA.
- **TCP/IP** содержит [таблицы TCP/IP](#), которые используются для конфигурирования связей текущего проекта IWS с другими рабочими станциями со средами исполнения IWS по протоколу TCP/IP.

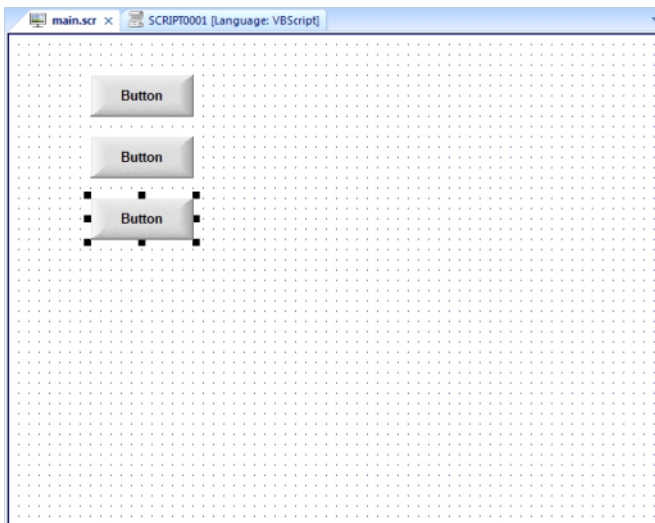
Клиентский и серверный модули IWS TCP/IP позволяют двум и более проектам синхронизировать свои базы данных по TCP/IP.

- **DDE** содержит [таблицы DDE](#), которые используются для конфигурирования DDE-клиента для DDE-серверных приложений, такие как Microsoft Excel или другие программы Windows, которые поддерживают этот интерфейс.

DDE (Dynamic Data Exchange) - протокол, который позволяет вести динамический обмен данными между Windows приложениями. IWS предоставляет серверный и клиентский DDE интерфейсы.

Редактор экранных форм (**Screen/Worksheet Editor**)

Этот мощный, объектно-ориентированный редактор предназначен для создания и изменения экранных форм и рабочих таблиц проекта. Экранные формы служат для ввода данных оператором (используя мышь и клавиатуру), а также для динамической визуализации данных проекта.



Редактор экранных форм

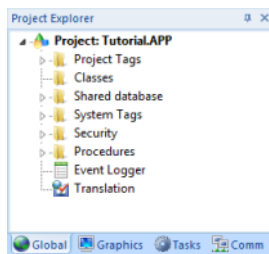
К ключевым особенностям редактора экранных форм можно отнести следующие:

- редактирование с использованием механизмов Point-and-click, drag-and-drop
- Группирование объектов с сохранением геометрического распределения индивидуальных объектов
- Редактирование объектов без разгруппирования на компоненты или группы
- Обработка растровых изображений и фоновых изображений
- Поддержка строки состояния в окнах проекта и диалоговых окнах

Информация о тегах и базе данных проекта

Тег – базовый компонент любого проекта IWS. Теги – это переменные, используемые в IWS для приема и хранения данных от драйверов устройств ввода/вывод (подключенного оборудования), результатов вычислений и функций, данных, введенных оператором. Теги используются при отображении информации на экранах и Web-страницах, для управления объектами на экране, для управления [исполняемыми задачами](#) (runtime tasks).

Но теги – это больше, чем просто переменные. IWS содержит подсистему управления базами данных в реальном масштабе времени, которая обеспечивает работу множества сложных функций, таких как сопровождение метками времени любых изменений значений, проверка соответствия значения тега области допустимых величин, анализ на предмет выхода значения тега за пороги аварийной сигнализации, и так далее. Теги IWS кроме параметра "значение" имеют множество иных параметров и свойств, одни из которых доступны в режиме разработки, другие – только в режиме исполнения.



Все теги структурированы в следующие категории, представленные папками в [секции Global](#) проводника проекта:

- [Теги проекта](#) (Project Tags) – теги, которые создаются во время разработки проекта. Подразделяются на:
 - Экранные теги
 - Теги ввода-вывода, через которые осуществляется обмен данными с драйверами подключенного оборудования
 - Управляющие теги
 - Вспомогательные теги, используемые для математических вычислений
- [Разделяемая база данных](#) (Shared Database) – теги, которые можно импортировать в IWS из проектов, разработанных в популярных

программах для программирования промышленных контроллеров и "SoftPLC".

Например, вы можете создать теги в SteepleChase и затем импортировать их в IWS, таким образом у IWS появится возможность вести обмен данными с системой SteepleChase.

Однако изменять такие теги в среде IWS невозможно. Для их редактирования необходимо воспользоваться их исходной средой разработки, а затем повторить импорт в базу тегов IWS.

- **Системные теги** (System Tags) - предопределенные теги, предназначенные для внутреннего использования в IWS. Например:

- Тег даты (Date) хранит текущую дату в строковом формате
- Тег времени (Time) хранит текущее время в строковом формате

Большинство системных тегов доступны только для чтения, т.е. их невозможно добавлять, редактировать или удалять из базы данных тегов.

Чтобы посмотреть полный список системных тегов, выберите секцию **Global** в проводнике проекта, откройте папку **System Tags**, затем откройте подпапку **Tag List**. На рисунке выше частично показан список системных тегов.

После создания тега вы можете использовать его в любом месте в пределах проекта, и каждый тег может быть использован одновременно в нескольких объектах или атрибутах объекта.

Правила написания имен тегов

При выборе имени тега руководствуйтесь следующими принципами:

- Имя тега должно быть уникальным: нельзя объявить два разных тега (или функции) с одним именем. Если вы попытаетесь создать тег с уже существующим именем, IWS распознает совпадение и не позволит создать новый тег.
- Имя тега должно начинаться с буквы. Далее могут следовать буквы, цифры, знаки подчеркивания (`_`).
- В именах тегов нельзя использовать следующие символы:

`` ~ ! @ # $ % ^ & * () - = \ + \ [] { } < > ?`

- Максимальная длина имени тега, как и максимальная длина имени класса тегов, составляет 255 символов. Можно использовать заглавные и прописные буквы. Имена тегов нечувствительны к регистру. Поскольку IWS не делает различий между символами верхнего и нижнего регистра, можно использовать оба варианта для придания именам тегов лучшей читаемости. (Например, `TankLevel` вместо `tanklevel`.)
- Имена тегов должны отличаться от имен системных тегов и математических функций.



Примечание: Использование символа `@` в начале имени тега говорит о том, что используется **косвенный тег** (indirect tag).

Примеры допустимых имен тегов:

- `Temperature`
- `pressure1`
- `count`
- `x`

Типы данных тегов

При создании тегов необходимо указать, к какому типу данных относится тег. IWS поддерживает следующие стандартные типы данных:

- **Boolean** (1 бит): Простой булевский тип данных с возможными значениями 0 (false) и 1 (true). Эквивалент "bool" типу данных в C++. Обычно используется для включения и выключения объектов, для открытия и закрытия объектов.
- **Integer** (4 байта): Целое число (положительное, отрицательное или ноль), имеет внутреннее 32-битное представление со знаком. Эквивалент "signed long int" типу данных в C++. Обычно используется для счета целых чисел или задания целочисленных величин. Пример: 0, 5, -200.
- **Real** (с плавающей запятой, 8 байт): Вещественное число, имеет внутреннее 64-битное представление со знаком. Эквивалент "double" типу данных в C++. Обычно используется для измерений или хранения дробных величин.
- **String** (буквенно-цифровые значения, до 1024 символов): Символьная строка длиной до 1024 символов, содержащая буквы, цифры, или специальные символы. Поддерживает ASCII и UNICODE символы. Пример: `Recipe product X123, 01/01/90, *** On ***`.

Также можно превратить тег в составной тег, объявив его как тег некоторого [класса](#). Класс – это шаблон, состоящий из нескольких объявлений тегов, каждый со своим типом данных. Классы можно использовать в проектах, имеющих объекты (например, емкости для жидкости) с множеством атрибутов (например, уровень жидкости, температура, давление) для контроля и управления.

Вы можете найти эти типы данных тегов (и их соответствующие значки) в [секции Global](#) проводника проекта.

Смотрите также: [Правила написания имен тегов](#)

Изменение механизма присвоения числовых значений **Boolean**-тегам

По умолчанию, если какое-либо числовое значение отличное от нуля (т.е. $\neq 0$) записывается в тег типа Boolean, тег автоматически принимает значение 1. Такое поведение, при необходимости, можно изменить, редактируя в файле `project_name.app` параметр:

```
[Options]
BooleanTrueAboveZero=value
```

Если для **BooleanTrueAboveZero** установлено значение по умолчанию 0, поведение проекта будет соответствовать описанному выше. Если для **BooleanTrueAboveZero** установлено 1, поведение проекта будет следующим:


- Если в Boolean-тег записывается значение меньше или равное 0 (т.е. ≤ 0), тег принимает значение 0 (false).
- Если в Boolean-тег записывается значение больше 0 (т.е. > 0), тег принимает значение 1 (true).



Внимание: Это глобальная настройка. Если вы хотите изменить механизм присвоения только для некоторых тегов, вы не должны менять эту опцию.

Массивы тегов


Тег IWS может содержать одно значение или массив значений.

 **Примечание:** Максимальный размер массива 16384 элемента. Дополнительно нужно учитывать ограничение по количеству тегов в лицензии. Каждый элемент массива (включая нулевой элемент) считается за отдельный тег по условиям лицензионного ограничения, потому что каждый элемент может содержать независимое значение.

Массив тегов – это ряд тегов с одинаковыми именами, идентифицируемыми с помощью индекса (матрица размером n строк на 1 столбец). Максимальный размер массива зависит от спецификации продукта. Для доступа к элементам массива используется следующий синтаксис:


ArrayTagName [ArrayIndex]

Например: **tank [0]**, **tank [1]**, **tank [2]** и **tank [500]**.

 **Внимание:** Вы должны задать максимальный индекс для каждого массива тегов в колонке **size** таблицы. Заданное число n соответствует количеству элементов массива от 0 до n . Например, если размер TagA 3, элементы могут быть **TagA [0]**, **TagA [1]**, **TagA [2]** и **TagA [3]**.

По возможности старайтесь использовать массив тегов, потому что это оптимизирует использование памяти и упрощает задачи конфигурирования. Например, если вы хотите отображать на мониторе емкость из нескольких однотипных, вы можете использовать массив тегов для конфигурирования единственного экрана, содержащего теги, которые можно связать с любой из емкостей. Например, используя тег **tk** в качестве индекса номера емкости: **pressure [tk]**, **temperature [tk]** и **temperature [tk+1]**.

Индекс массива может быть тегом, числом, выражением, содержащим арифметический оператор "+".

 **Примечание:** При обращении к элементу массива по индексу, содержащему арифметический оператор +, следует использовать следующий синтаксис:

ArrayTagName [NumValue1+NumValue2]


Где *NumValue1* и *NumValue2* могут быть целочисленными тегами или числовыми константами. Например: `temperature[tk+2]` или `temperature[tk+6]`.

Использование массивов тегов в IWS может значительно сократить время разработки проекта. Например, вам необходим тег для точек температур 4-х емкостей. Стандартный способ следующий:

- **temperature1**: верхняя температура емкости 1
- **temperature2**: верхняя температура емкости 2
- **temperature3**: верхняя температура емкости 3
- **temperature4**: верхняя температура емкости 4

С использованием массива тегов решение задачи упрощается:

- **temperature[j]**: верхняя температура емкости j

 Примечание: Когда вы создаете 4-х элементный массив тегов, система создает 5 позиций (от 0 до 4). Например:

```
tag_example[15] //start position=0, end position=15
```

Следовательно, `tag_example[15]` – массив из 16-и элементов.

Если, при использовании другого тега в качестве индекса массива, значение этого тега выходит за границы размера массива, то это приводит к следующему результату:

- Если *IndexTag* больше чем размер массива, `MyArray[IndexTag]` укажет на последний элемент массива;
- Если *IndexTag* меньше 0, `MyArray[IndexTag]` укажет на начальный элемент массива.

Массив тегов

Массив тегов состоит из ряда тегов, которые имеют одинаковые имена, но обладают уникальными индексами (матрица размером *n* строк на 1 столбец) для дифференциации каждого тега. Индекс массива может быть фиксированной величиной, другим тегом или выражением. Максимальный размер массива определяется спецификацией продукта.

Массив тегов используется для:

- Облегчения конфигурирования

- Возможности переключения экранов, рецептов, коммуникационных интерфейсов
- Экономии времени разработки при объявлении тегов

Объявление массива тегов возможно в двух форматах:

- Для простого массива тегов:

ArrayTagName [ArrayIndex]

- Для комплексного массива тегов (где индекс массива представляет собой выражение, состоящее из другого тега и математической операции):

ArrayTagName [ArrayIndex+c]

Где:

- **ArrayTagName** – имя тега;
- **[ArrayIndex]** – уникальный индекс (фиксированное значение или другой тег);
- **+** - арифметическая операция;
- **c** - числовая константа.

 **Примечание:**

- Необходимо определить максимальный индекс для каждого массива тегов, задавая значение (*n*) в колонке *Array Size* таблицы *Project Tags* или в поле *Array Size* диалогового окна *New Tag*. (Смотрите [Creating project database Tags](#)).

Когда вы создаете *n*-позиционный массив тегов, IWS фактически создает ***n*+1** позицию (от 0 до *n*). Например, если вы определите **ArrayTag[15]**, массив будет иметь 16 элементов, где 0 - начальная позиция, 15 - конечная позиция.

- Не используйте пробелы в именах элементов массивов тегов.

IWS считывает тег, начиная с первого символа, и продолжает до тех пор, пока не будет достигнут первый пробел или нулевой символ. Следовательно, система не распознает символы, следующие за пробелом, в качестве части записи массива тегов.

Например, если вы напишете **a[second + 1]**, IWS попытается распознать "a[second" как тег и посчитает это ошибкой. Если вы

запишите `a[second+1]`, то это будет правильной записью имени элемента массива тегов.

Вы можете использовать массив тегов везде, где можно использовать обычные теги. Также массивы тегов значительно упрощают конфигурирование и могут сэкономить время разработки, поэтому мы рекомендуем использовать их по возможности.

Например, предположим, что вы хотите отслеживать температуру 4 емкостей. При традиционном способе конфигурирования это может выглядеть так:

- `temperature1` — верхняя температура емкости 1
- `temperature2` — верхняя температура емкости 2
- `temperature3` — верхняя температура емкости 3
- `temperature4` — верхняя температура емкости 4

Вы можете использовать массивы тегов для упрощения этой задачи следующим образом (n – количество емкостей):

- `temperature[n]` — верхняя температура емкости n

Следующая таблица содержит дополнительные примеры использования массива тегов:

Примеры определения массивов тегов

Пример массива тегов	Описание
<code>Tank [1], Tank [2], Tank [500]</code>	Простой массив с индексами (1, 2 и 500), которые являются числовыми константами. Например, номера емкостей.
<code>Tank [tk]</code>	Простой массив, у которого индекс <code>tk</code> является тегом. Например, тег содержит номер емкости.
<code>Tank [tk+1]</code>	Комплексный массив, у которого индекс (<code>tk+1</code>) является выражением. Например, <code>tk</code> (номер емкости) + 1.



Примечание: При использовании другого тега в качестве индекса массива, если значение тега выходит за границы размера массива, это приводит к следующему результату:

- Если `IndexTag` больше чем размер массива, `MyArray[IndexTag]` укажет на последний элемент массива;

- Если *IndexTag* меньше 0, **MyArray[IndexTag]** укажет на начальный элемент массива (т.е. **MyArray[0]**).

Косвенные теги (indirect tags)

IWS поддерживает косвенный доступ к тегам. Например, рассмотрим тег **X** строкового типа. Этот тег может хранить имя любого другого тега (таким образом, он может "указывать" на другой тег любого типа, включая класс). Синтаксис для косвенного обращения к тегам следующий: `@IndirectTagName`. Например, предположим, что тег **X** содержит строку "**TEMP**". Чтение/запись в `@X` приводит к изменению тега **TEMP**.



Примечание: Любой тег строкового типа потенциально может быть использован в качестве косвенного тега.

Чтобы обратиться к тегу типа класс, вы можете объявить строковый тег, который будет указывать на тег типа класс. Например:

Класс	TANK с членом Level
Тег	TK типа класс TANK
Тег	XCLASS строкового типа

Чтобы получить доступ к значению **TK.Level** следует записать строку "**TK.Level**" в тег **XCLASS** и затем использовать синтаксис `@XCLASS`. Можно также обратиться к члену класса напрямую, объявив другой тег типа класс, который будет указывать на первый тег типа класс.

Например:

Класс	TANK с членом Level
Тег	TK типа класс TANK
Тег	XCLASS типа класс TANK

Чтобы получить доступ к значению **TK.Level** следует записать строку "**TK**" в тег **XCLASS**, затем использовать синтаксис `@XCLASS.Level`.

При объявлении косвенного тега, поместите `@XCLASS` в колонку имени тега, вместо того чтобы создавать строковый тег. Тип тега выбирается в соответствии с типом тега, на который будет ссылаться новый. Примеры: `@z: Integer`, `@XClass: TANK`.

Косвенные теги

Косвенные теги "указывают" на другие теги (включая теги типа класс). Использование косвенных тегов может сэкономить время разработки, потому что они заменяют дублирование тегов (и связанной с ними логики).

Вы можете делать косвенный тег из любого строкового тега, просто добавляя впереди имени тега символ @ (при выводе), @*TagName*.

- Косвенное обращение к простому тегу. Предположим, строковый тег `strX` содержит строку "`Tank`", имя другого тега. Тогда, при обращении по `@strX`, получаем доступ к значению тега `Tank`.
- Косвенное обращение к членам класса. Создается строковый тег, который указывает на член класса. Например, строковый тег `strX` содержит строку "`Tank.Level`", являющуюся полным именем члена класса. Тогда, при обращении по `@strX`, получаем доступ к члену `Tank.Level`.
- Прямое обращение к члену класса с помощью косвенного обращения к тегу типа класс. Например, член класса `Tank.Level`. Необходимо создать (объявить) тег типа класс `@XCLASS` (со знаком @ вначале имени тега), записать в него строку "`Tank`". Для доступа использовать синтаксис `@XCLASS.Level`.

СПИСОК СВОЙСТВ ТЕГОВ

Свойства тега (также известные как поля тега) – метаданные, присоединенные к каждому тегу. Большинство из этих свойств может быть установлено с помощью диалогового окна свойства тега (*Tag Properties*), которое открывается по щелчку на кнопке **Tag Properties** в панели инструментов.

Для доступа к свойствам тега в режиме исполнения используется следующий синтаксис (без пробелов), в любом месте, где можно применять обычные теги:

```
tag_name->property_name
```

В режиме исполнения можно получить доступ к следующим свойствам тега:

Свойство тега	Описание	R или R/W	Тип данных	Доступен для данных типа...				Сохраняется
				Bool	Int	Real	Str	
Name	Имя тега из базы тегов проекта.	R	String, до 32 символов	Y	Y	Y	Y	n/a
MemberName	Имя члена класса . Примечание: Синтаксис: Class.Member->MemberName Пример: Tank.Lvl->MemberName = Lvl	R	String, до 32 символов	Y	Y	Y	Y	n/a
Size	Размер массива. Если тег не является массивом, возвращает 0	R	Integer	Y	Y	Y	Y	n/a
Index	Индекс элемента массива . (Массив - любой тег с size больше 0) Примечание: Синтаксис: Tag[Index] ->Index	R	Integer	Y	Y	Y	Y	n/a

Свойство тега	Описание	R или R/W	Тип данных	Доступен для данных типа...				Сохра
				Bool	Int	Real	Str	
	Пример: Tag [1] - >Index = 1							
Description	Описание тега, задаваемое в таблице тегов.	R	String	Y	Y	Y	Y	Y
Quality	<p>Качество тега (192=GOOD; 0=BAD). Проект обновляет это поле каждый раз, когда обновляется значение тега в результате вычислений или в результате операции ввода-вывода (например, от драйвера OPC).</p> <p>Если результат недопустим (такой как деление на 0), или возникает ошибка ввода-вывода, связанная с этим тегом, то параметр "quality" устанавливается в "BAD".</p>	R	Integer	Y	Y	Y	Y	N
TimeStamp	Время и дата, соответствующие последнему изменению значения тега.	R	String	Y	Y	Y	Y	N
Blocked	<p>Данное свойство может принимать два значения:</p> <ul style="list-style-type: none"> 0: Тег заблокирован, и все выполняемые задачи игнорируют его. Эффективно удаляется из базы данных проекта. 1: Тег разблокирован, и все выполняемые задачи имеют к нему доступ. <p>Это свойство используется, когда</p>	R/W	Boolean	Y	Y	Y	Y	N

Свойство тега	Описание	R или R/W	Тип данных	Доступен для данных типа...				Сохраняется
				Bool	Int	Real	Str	
	необходимо динамически отключить все действия, связанные с тегом. Даже когда тег заблокирован, он учитывается в тегах лицензии.							
Unit	Краткое описание (до 9 символов) технических единиц измерения тега. Например, Kg, BTU, psi.	R/W	String, до 9 символов	Y	Y	Y	Y	Y
Max	Максимальное значение, которое может быть записано в тег в режиме исполнения.	R/W	Real	N	Y	Y	N	Y
Min	Минимальное значение, которое может быть записано в тег в режиме исполнения.	R/W	Real	N	Y	Y	N	Y
B0 ... B31	Значение (0 или 1) любого из 32 бит (b0, b1, b2, ... b31) тега типа Integer. (B0: LSB B31: MSB)	R/W	Boolean	N	Y	N	N	N
DisplayValue	<p>Преобразованная величина тега, только для отображения на экране:</p> $\text{DisplayValue} = (\text{Value} / \text{UnitDiv}) + \text{UnitAdd}$ <p>Используется, когда текущая величина тега имеет одни технические единицы (см. выше про Unit), а требуется отобразить на экране в других единицах (см. ниже DisplayUnit). Например, градусы Цельсия и градусы Фаренгейта.</p> <p>Если пользователь изменяет (вводит) DisplayValue в режиме</p>	R/W	Real	N	Y	Y	N	n/a

Свойство тега	Описание	R или R/W	Тип данных	Доступен для данных типа...				Сохра
				Bool	Int	Real	Str	
	исполнения, перед записью в тег величина преобразуется обратно: Value = (DisplayValue - UnitAdd) * UnitDiv							
DisplayUnit	Короткое описание (до 9 символов) технических единиц измерения DisplayValue . Примечание: Это свойство может быть установлено только с помощью функций SetDisplayUnit и SetTagDisplayUnit .	R	String, до 9 символов	N	Y	Y	N	N
UnitDiv	Число, на которое делится величина тега для получения DisplayValue . Если деление не нужно, UnitDiv должно быть равно 1. Примечание: Это свойство может быть установлено только с помощью функций SetDisplayUnit и SetTagDisplayUnit .	R	Real	N	Y	Y	N	N
UnitAdd	Число, которое прибавляется к значению тега для получения DisplayValue . Если добавка не требуется, UnitAdd должно быть равно 0. Примечание: Это свойство может быть установлено только с помощью функций SetDisplayUnit и SetTagDisplayUnit .	R	Real	N	Y	Y	N	N
DisplayMax	Максимальное значение, которое может быть	R/W	Real	N	Y	Y	N	N

Свойство тега	Описание	R или R/W	Тип данных	Доступен для данных типа...				Сохраняется
				Bool	Int	Real	Str	
	<p>введено в DisplayValue режиме исполнения:</p> $\text{DisplayMax} = (\text{Max} / \text{UnitDiv}) + \text{UnitAdd}$ <p>Если DisplayMax изменяется в режиме исполнения, Max также изменяется следующим образом:</p> $\text{Max} = (\text{DisplayMax} - \text{UnitAdd}) * \text{UnitDiv}$							
DisplayMin	<p>Минимальное значение, которое может быть введено в DisplayValue в режиме исполнения:</p> $\text{DisplayMin} = (\text{Min} / \text{UnitDiv}) + \text{UnitAdd}$ <p>Если DisplayMin изменяется в режиме исполнения, Min также изменяется следующим образом:</p> $\text{Min} = (\text{DisplayMin} - \text{UnitAdd}) * \text{UnitDiv}$	R/W	Real	N	Y	Y	N	N
HiHiLimit	Граница для HiHi тревоги.	R/W	Real	N	Y	Y	N	Y
HiLimit	Граница для Hi тревоги.	R/W	Real	N	Y	Y	N	Y
LoLimit	Граница для Lo тревоги.	R/W	Real	N	Y	Y	N	Y
LoLoLimit	Граница для LoLo тревоги.	R/W	Real	N	Y	Y	N	Y
RateLimit	Граница для тревоги по скорости (Rate).	R/W	Real	N	Y	Y	N	Y
DevSetpoint	Уставка для тревоги по отклонения (Deviation).	R/W	Real	N	Y	Y	N	n/a
DevPLimit	Граница для Deviation+ тревоги.	R/W	Real	N	Y	Y	N	Y
DevMLimit	Граница для Deviation- тревоги.	R/W	Real	N	Y	Y	N	Y

Свойство тега	Описание	R или R/W	Тип данных	Доступен для данных типа...				Сохраняется
				Bool	Int	Real	Str	
HiHi	Если 0, HiHi тревога не активна. Если 1, HiHi тревога активна.	R	Boolean	Y	Y	Y	N	n/a
Hi	Если 0, Hi тревога не активна. Если 1, Hi тревога активна.	R	Boolean	Y	Y	Y	N	n/a
Lo	Если 0, Lo тревога не активна. Если 1, Lo тревога активна.	R	Boolean	Y	Y	Y	N	n/a
LoLo	Если 0, LoLo тревога не активна. Если 1, LoLo тревога активна.	R	Boolean	Y	Y	Y	N	n/a
Rate	Если 0, Rate тревога не активна. Если 1, Rate тревога активна.	R	Boolean	Y	Y	Y	N	n/a
DevP	Если 0, Deviation+ тревога не активна. Если 1, Deviation+ тревога активна.	R	Boolean	N	Y	Y	N	n/a
DevM	Если 0, Deviation- тревога не активна. Если 1, Deviation- тревога активна.	R	Boolean	N	Y	Y	N	n/a
AlrStatus	<p>Integer величина, содержащая состояние текущих тревог тега. Каждый бит величины отражает активность соответствующей тревоги:</p> <ul style="list-style-type: none"> • Bit 0 (LSB): HiHi тревога активна • Bit 1: Hi тревога активна • Bit 2: Lo тревога активна • Bit 3: LoLo тревога активна • Bit 4: Rate тревога активна 	R	Integer	Y	Y	Y	N	N

Свойство тега	Описание	R или R/W	Тип данных	Доступен для данных типа...				Сохраняется
				Bool	Int	Real	Str	
	<ul style="list-style-type: none"> Bit 5: Deviation+ тревога активна Bit 6: Deviation- тревога активна <p>Пример: Если Tag->AlrStatus возвращает значение 2, это означает, что "H" тревога активна. Если возвращает значение 3, это означает, что "Hi" и "H" тревоги активны одновременно.</p> <p>Если свойство возвращает 0, это означает, что в данный момент нет активных тревог, связанных с тегом.</p> <p>Для Boolean-тегов, только значения: 1 (bit1), 4 (bit2) или 16 (bit4) могут быть возвращены.</p>							
Ack	<p>Это свойство имеет два значения:</p> <ul style="list-style-type: none"> 0: Когда нет тревог, связанных с данным тегом, которые требуют подтверждения. 1: Когда есть одна и более тревог, связанных с тегом, которые требуют подтверждения. <p>Является общим для всех тревог тега и сбрасывается в 0, только когда все тревоги подтверждены.</p>	R	Boolean	Y	Y	Y	N	N

Свойство тега	Описание	R или R/W	Тип данных	Доступен для данных типа...				Сохраняется
				Bool	Int	Real	Str	
UnAck	<p>Это свойство имеет два значения:</p> <ul style="list-style-type: none"> 0: Существует, по крайней мере, одна тревога, требующая подтверждения. 1: Нет тревог, требующих подтверждения. <p>Если вручную записать значение 1, то активные тревоги (если есть) будут подтверждены. Значение этого свойства всегда противоположно Ack.</p>	R/W	Boolean	Y	Y	Y	N	N
AlrAckValue	<p>Текст, связанный с подтвержденным состоянием булевого тега. Текст отображается в колонке Value Alarm/Event Control.</p> <p>Этот текст также можно редактировать в диалоговом окне Tag Properties (тревоги типа Bool).</p>	R/W	String, до 32 символов	Y	N	N	N	Y
AlrOffValue	<p>Текст, связанный с нормализованным состоянием булевого тега. Текст отображается в колонке Value Alarm/Event Control.</p> <p>Этот текст также можно редактировать в диалоговом окне Tag Properties (тревоги типа Bool).</p>	R/W	String, до 32 символов	Y	N	N	N	Y
AlrOnValue	<p>Текст, связанный с активным состоянием булевого тега. Текст</p>	R/W	String, до 32 символов	Y	N	N	N	Y

Свойство тега	Описание	R или R/W	Тип данных	Доступен для данных типа...				Сохраняется
				Bool	Int	Real	Str	
	<p>отображается в колонке Value Alarm/Event Control.</p> <p>Этот текст также можно редактировать в диалоговом окне Tag Properties (тревоги типа Bool).</p>							
AlrDisable	<p>Это свойство имеет два значения:</p> <ul style="list-style-type: none"> 0: Тревоги, связанные с тегом, отслеживаются (enabled). Это означает, что при возникновении события тревога активируется. 1: Тревоги, связанные с тегом, не отслеживаются (disabled). Это означает, что при возникновении события, тревога не активируется. 	R/W	Boolean	Y	Y	Y	N	N



Примечание:

- Если в колонке "сохраняемость" для свойства присутствует "п/а", это означает, что свойство напрямую зависит от объявления тега (например, Name, Size), или значение свойства непрерывно вычисляется в режиме исполнения (например, активация тревоги, DisplayValue). Чтобы включить сохраняемость для тега, выберите опцию **Retentive Parameters** в [диалоге Tag Properties](#).
- Если проект пытается записать значение, лежащее за пределами диапазона, определенного свойствами **Min** и **Max**, база тегов не примет новое значение, и в окне [Output](#) будет сформировано предупреждающее сообщение. Если значения свойств **Min** и **Max** равны 0, это означает, что любое возможное для данного типа тега значение может быть присвоено.

- Нельзя использовать свойства тегов (такие как битовые поля) для конфигурирования таблиц *Alarm* или *Trend*.
- Можно применять свойства тегов к **системным тегам**, но эти свойства не будут сохраняться при загрузке проекта на устройство СЕ.

Сброс базы данных тегов

Выберите команду **Reset Tags Database** для "перезагрузки" базы тегов на локальной рабочей станции. Эта команда влияет на все теги, хранящиеся в папке *Project Tags*. Команда может быть использована в случае, если необходимо "сбросить" текущие значения тегов проекта и восстановить значения "по умолчанию", которые они имели сразу после загрузки проекта. Когда вы останавливаете проект, но оставляете среду разработки открытой, теги не сбрасываются при повторном запуске. Поэтому, если это необходимо, то следует выполнять команду сброса перед повторным запуском проекта.

Когда выполняется эта команда, начальные значения, сконфигурированные для каждого тега (*диалог Tags Properties*) записываются в соответствующие теги. Если начальные значения не сконфигурированы для числовых тегов (**Boolean**, **Integer**, **Real**), присваивается значение 0. Если начальные значения не сконфигурированы для строковых тегов, присваивается пустая строка ("").

Эта команда заблокирована (disabled) (серая), если, по меньшей мере, одна runtime-задача выполняется на локальной рабочей станции. Необходимо закрыть все задачи (**Stop** на вкладке Home), прежде чем эта команда сможет быть выполнена.



Примечание: Теги, хранящиеся в папке *System Tags* и в *Shared Tags*, (если есть) не сбрасываются по этой команде.



Подсказка: Если необходимо, чтобы теги сбрасывались автоматически, всякий раз, когда проект запускается на исполнение (**Run** на вкладке Home), можно активировать опцию сброс базы данных тегов при запуске проекта на вкладке *Preference* диалогового окна *Project Setting*.

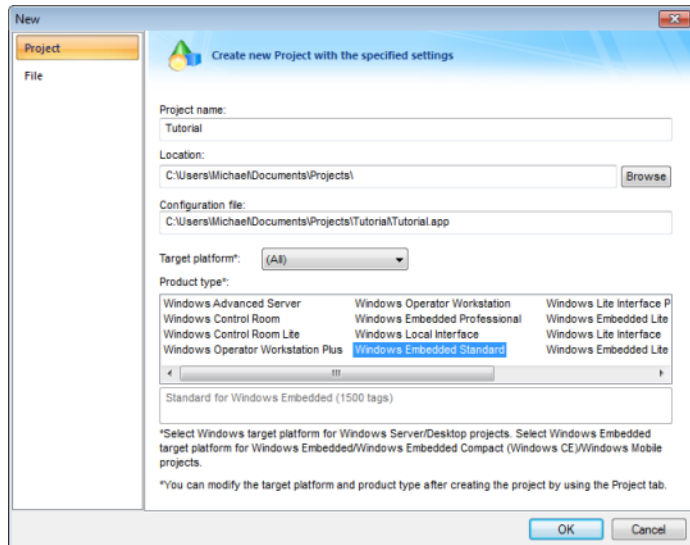
Учебное руководство: Создание демонстрационного проекта

В этом разделе шаг за шагом объясняется, как разработать простой проект и сконфигурировать драйвер ввода/вывода.

Создание нового проекта

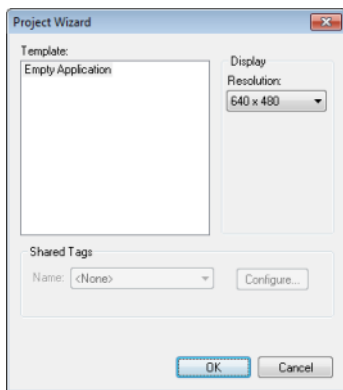
В этой части руководства показано, как создается новый проект, включая выбор целевой платформы.

1. Щелкните кнопку приложения в верхнем левом углу среды разработки и выберите *New* в меню приложения. Отобразится диалоговое окно *New*.
2. Щелкните по вкладке **Project**.
3. В поле **Project name** введите название вашего проекта. В этом руководстве выбрано название `Tutorial`. Среда разработки автоматически создаст новый каталог с таким же именем и разместит файл проекта в этой директории. (Обратите внимание на текстовое поле **Configuration file** на рисунке ниже.) Чтобы разместить проект в месте отличном от стандартной директории проектов, щелкните **Browse** и переместитесь в другое место.
4. В списке **Product type** выберите тип проекта, который вы хотите создать. Для нашего примера выберите **Windows Embedded Standard**. Это функционально-ограниченный тип продукта, который может быть успешно "развернут" на устройстве Windows Embedded.



Выбор целевой платформы и типа продукта

5. Щелкните **OK**.
Диалог *New* закроется, следом отобразится диалог *Project Wizard*.
6. В списке шаблонов (**Template**) выберите пустой проект (**Empty Application**).
7. Из списка **Resolution** выберите разрешение **640 x 480**.



Выбор шаблона пустого проекта и экранного разрешения **640x480**

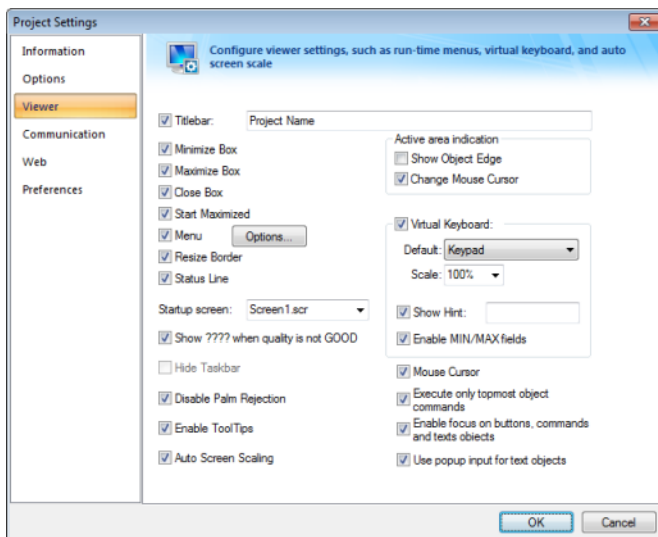
8. Щелкните **OK**.
Диалог *Project Wizard* закроется, и в среде разработки будет создан новый проект.

Установка стартового экрана

В этой части руководства рассказывается, как открыть настройки проекта и задать экран, который будет отображаться при запуске проекта.

- Секция **Information** содержит информацию о проекте (описание проекта, номер ревизии, имя компании, имя автора, описание подключаемого оборудования, примечания).
 - Секция **Options** определяет общие настройки проекта, такие как Target System, Automatic Translation, Alarm history and Events, Default Database и Shared Tags.
 - Секция **Viewer** задает общие экранные параметры системы исполнения.
 - Секция **Communication** определяет общие коммуникационные параметры, связанные с проектом.
 - Секция **Web** определяет настройки Web-решений, такие как Data Server IP-адрес.
 - Секция **Preferences** настраивает предупреждающие сообщения.
1. На вкладке **Project** ленты, в группе **Settings** щелкните **Viewer**.
Отобразится диалог *Project Settings* с выбранной секцией **Viewer**.

2. В поле **Startup screen** задайте экран `main.scr`.



Установка стартового экрана

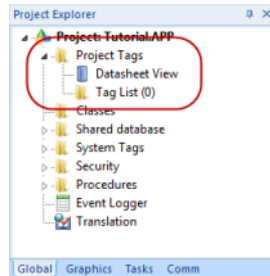
Когда вы запустите проект, этот экран автоматически отобразится первым. Вы можете задать экран перед тем, как он будет создан.

3. Щелкните **OK**.

Создание тегов

В этой части руководства рассказывается, как создать новые теги, путем добавления их в таблицу Project Tags.

Тег - любая переменная, которая содержит некоторое значение. Все теги, созданные в проекте, содержатся в папке Project Tags в секции Global проводника проекта.



Папка с тегами проекта

1. В проводнике проекта щелкните по секции **Global**.
2. Двойной щелчок по **Project Tags** раскроет папку.
3. Двойной щелчок по **Datashheet View** откроет таблицу *Project Tags*.
4. Используйте следующие параметры при создании тегов для демонстрационного проекта.
 - a) **Name**: Задайте уникальной имя тега. Для данного проекта задайте `Level` (уровень).
 - b) **Array**: Задайте верхний индекс массива тегов. (Для обычных тегов в колонке `Array` должен быть 0.) Для данного проекта задайте 3.

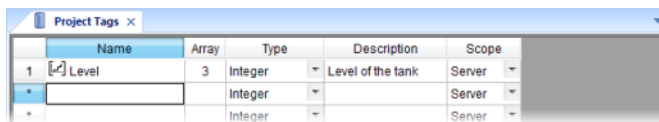
Каждый элемент массива будет соотноситься с одной из трех емкостей:

- `Level [1]` - уровень в емкости 1
- `Level [2]` - уровень в емкости 2
- `Level [3]` - уровень в емкости 3

Мы не используем имя `Level [0]` в этом проекте, хотя это имя также доступно в качестве тега.

- c) **Type**: Задайте тип данных тега: Boolean, Integer, Real, String или Class. Для данного проекта задайте **Integer**.
- d) **Description** (не обязательно): Вы можете задать описание тега для облегчения понимания и документирования проекта.
- e) **Scope**: Задайте область видимости тега (распределение между сервером и клиентской станцией).
 - **Local** выбирается, если вам требуется, чтобы тег имел разные независимые значения на сервере и клиентской рабочей станции.
 - **Server** выбирается, если вам требуется, чтобы тег содержал одинаковые данные на сервере и клиентской станции.

Для данного проекта выберите **Server**.



	Name	Array	Type	Description	Scope
1	[?] Level	3	Integer	Level of the tank	Server
*			Integer		Server
*			Integer		Server

Создание тега "уровня"

5. Сохраните и закройте таблицу *Project Tags*.

Дополнительные теги будут созданы при дальнейшем построении проекта.

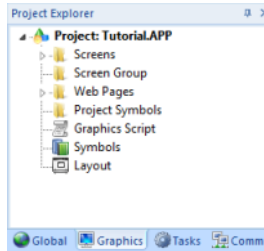


Подсказка: Можно сортировать данные в таблице *Project Tags*, добавлять, убирать дополнительные колонки, выбирать опции с помощью всплывающего по правому щелчку мыши меню.

Создание стартового экрана

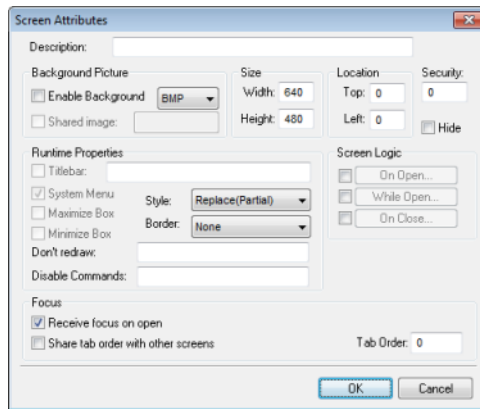
В этой части руководства рассказывается, как создать первый экран проекта, который будет содержать одну кнопку, открывающую другой экран.

1. В проводнике проекта щелкните по секции **Graphics**.



2. Щелкните правой кнопкой мыши по **Screens** и выберите **Insert** в контекстном меню.

Среда разработки хранит экраны проекта в папке **Screens**.
Отобразится диалог *Screen Attributes*.

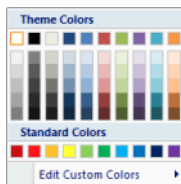


Диалоговое окно *Screen Attributes*

3. Используйте этот диалог для задания свойств экрана, таких как размер и тип экрана.

Для данного проекта щелкните **OK** для принятия настроек по умолчанию.
Диалог *Screen Attributes* закрывается, и откроется новый экран для редактирования в рабочем пространстве.

4. На вкладке **Graphics** ленты, в группе **Screen** щелкните на **Background Color**. Отобразится стандартная палитра цветов.
5. В палитре выберите светло-серый цвет.



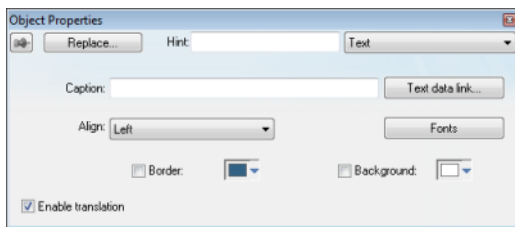
Палитра цветов

Фон экрана окрасится в выбранный цвет.

Создание заголовка стартового экрана

В этой части руководства рассказывается, как создать заголовок стартового экрана с использованием текстового объекта.

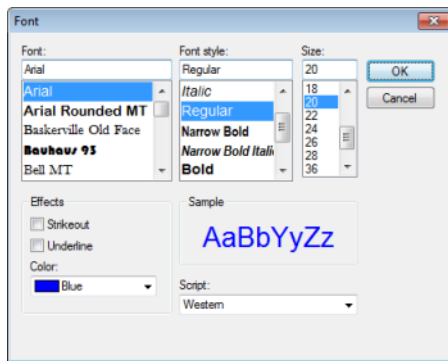
1. На вкладке **Graphics** ленты, в группе **Active Objects** щелкните **Text**. Курсор мыши поменяет свое изображение со стрелки на перекрестие.
2. Щелкните по экрану и введите "Welcome to the Tutorial Application", затем нажмите Return. Это действие создаст новый текстовый объект с заданным текстом.
3. Двойной щелчок по объекту откроет диалог свойств объекта *Object Properties*.



Object Properties: диалог создания текстового объекта

- Двойной щелчок на любом экранном объекте открывает диалог свойств объекта *Object Properties*. Свойства, отображаемые в диалоговом окне, зависят от типа объекта.

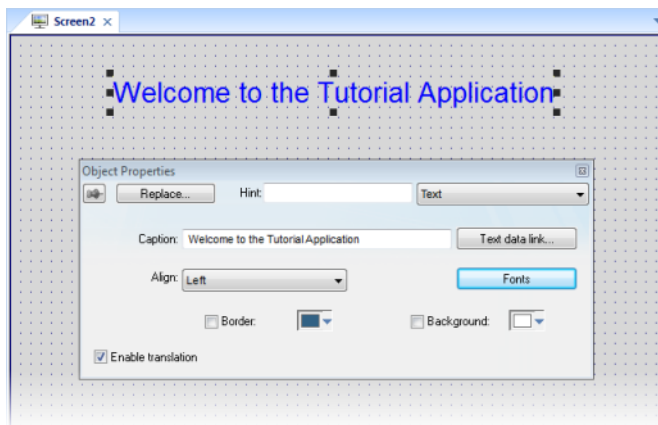
- Диалог *Object Properties* также содержит "канцелярскую" кнопку, с помощью которой диалог фиксируется в данной точке экрана. Кнопка переключает свое состояние при каждом нажатии:
 - Когда кнопка не нажата, фокус передается объекту на экране, как только он выбирается. Рекомендуется держать кнопку отжатой, когда работаете с объектами (копируете, вставляете, вырезаете и удаляете). Хотя диалог *Object Properties* присутствует на переднем плане, команды клавиатуры (**Ctrl+C**, **Ctrl+V**, **Ctrl+X** или **Del**) посылаются к объектам.
 - Когда кнопка нажата, диалог *Object Properties* сохраняется в фокусе, даже когда вы щелкаете по объектам. Мы рекомендуем держать кнопку нажатой, когда вы хотите менять настройки объектов. Вы можете щелкнуть по объекту и задать новое значение свойства в диалог *Object Properties* (нет необходимости щелкать по окну для перевода фокуса на него). Также, когда кнопка нажата, диалог *Object Properties* не закрывается автоматически, когда вы щелкаете по экрану.
4. Щелкните **Fonts** для открытия диалога выбора шрифта. Для данного проекта выберите:
- Шрифт: **Arial**
 - Начертание: **Regular**
 - Размер: **20**
 - Цвет: **Blue**



Настройка шрифта

5. Щелкните **OK** для закрытия диалога *Font*.

Настройки шрифта будут применены к текстовому объекту.



Настройка шрифта текстового объекта

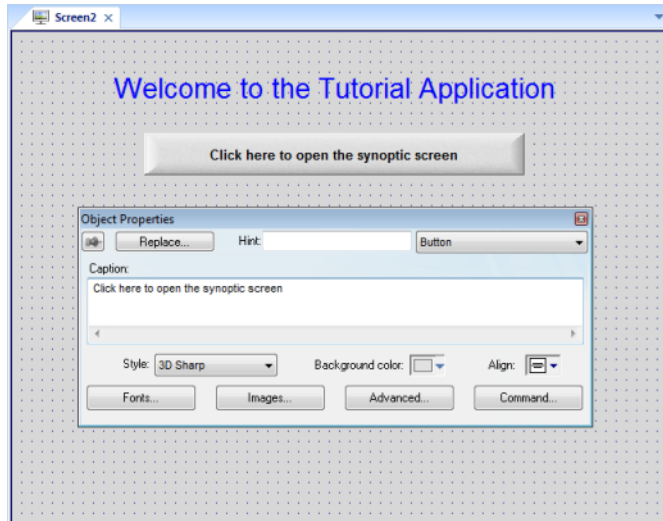
6. Закройте диалог *Object Properties* (закрывающая кнопка в правом верхнем углу).

Рисование кнопки для открытия другого экрана

В этой части руководства рассказывается, как нарисовать и сконфигурировать кнопку, которая будет открывать другой экран.

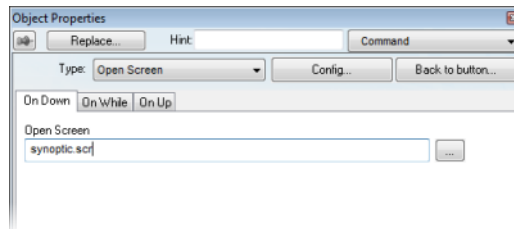
1. На вкладке **Graphics** ленты, в группе **Active Objects** щелкните по **Button**. Курсор мыши поменяет изображение со стрелки на перекрестие.
2. Щелкните по экрану и, удерживая кнопку в нажатом состоянии, двигайте курсор для рисования объекта – кнопки.
3. Двойной щелчок по объекту откроет диалог свойств объекта *Object Properties*.

4. В поле **Caption** (подпись) задайте следующий текст: "Нажмите сюда для открытия мнемосхемы".



Добавление подписи кнопки

5. Щелкните **Command**.
Диалог *Object Properties* отобразит свойства для конфигурирования команд.
6. Из списка **Type** выберите **Open Screen**.
7. В поле **Open Screen** введите `synoptic.scr`.



Конфигурирования команды открытия экрана

Вы можете задать экран, который пока не создан.

8. Закройте диалог *Object Properties*.

Сохранение и закрытие стартового экрана

В этой части руководства рассказывается, как правильно сохранить и закрыть экран.

1. Нажмите кнопку приложения (Application button) в левом верхнем углу среды разработки и щелкните **Save** в меню приложения. Отобразится стандартный диалог сохранения (*Save*) Windows.
2. В поле **File name** введите `main.scr`.
3. Щелкните **Save**.
Файл сохранится в папку проекта (`\project_name\Screen\main.scr`), и диалог *Save* закроется.
4. Щелкните кнопку приложения в левом верхнем углу среды разработки, затем щелкните **Close**.

Создание экрана мнемосхемы

В этой части руководства рассказывается, как создать второй экран проекта, который будет включать анимированную емкость с жидкостью и некоторые базовые средства управления.

1. В проводнике проекта щелкните по секции **Graphics**.
2. Щелкните правой кнопкой мыши по папке **Screens** и выберите **Insert** в контекстном меню.
Отобразится диалог *Screen Attributes*.
3. Используйте этот диалог для задания свойств экрана, таких как размер и тип экрана.
Для данного проекта щелкните **OK** для принятия настроек по умолчанию.
4. Щелкните кнопку приложения в левом верхнем углу среды разработки, затем щелкните **Save** в меню приложения.
Отобразится стандартный диалог сохранения (*Save*) Windows.
5. В поле **File name** введите `synoptic.scr`.
6. Щелкните **Save**.
Файл сохранится в папку проекта (`\project_name\Screen\synoptic.scr`), и диалог *Save* закроется.

Рисование заголовка экрана мнемосхемы

В этой части руководства рассказывается, как нарисовать заголовок экрана мнемосхемы с использованием текстового объекта.

1. На вкладке **Graphics** ленты, в группе **Active Objects** щелкните **Text**.
2. Щелкните по экрану и введите "Synoptic Screen", затем нажмите Return.
3. Двойной щелчок по объекту откроет диалог свойств объекта *Object Properties*.
4. Щелкните **Fonts** для открытия диалога выбора шрифта.
Для данного проекта выберите:
 - Шрифт: **Arial**
 - Начертание: **Bold**
 - Размер: **2**
 - Цвет: **Blue**
5. Закройте диалог *Object Properties*.

6. Переместите текстовый объект в верхний левый угол экрана.
7. Нажмите кнопку приложения (Application button) в левом верхнем углу среды разработки и щелкните **Save** в меню приложения.

Этот рисунок показывает, как должен выглядеть экран после этих действий.

Synoptic Screen

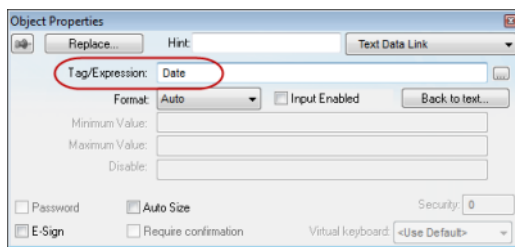
Заголовок экрана

Отображение текущего времени и даты

В этой части руководства рассказывается, как создать индикаторы времени и даты и связать их с системными тегами.

Date и **Time** – системные теги, которые содержат текущую дату и время локальной рабочей станции. Эти теги доступны в любом проекте.

1. На вкладке **Graphics** ленты, в группе **Active Objects** щелкните **Text**.
2. Щелкните по экрану и введите "Date: #####", затем нажмите Return.
3. Двойной щелчок по объекту откроет диалог свойств объекта *Object Properties*.
4. Щелкните **Text Data Link**.
Диалог *Object Properties* изменится для отображения свойств Text Data Link.
5. В поле **Tag/Expression** введите Date.

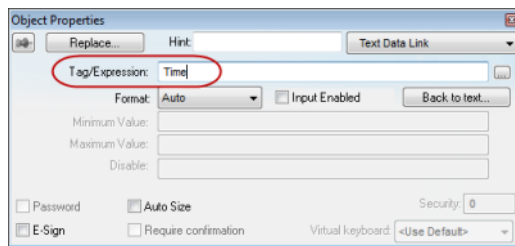


Задание тега **Date**

В режиме исполнения система исполнения заменяет символы ##### текстового объекта на значения системного тега **Date**.

6. Закройте диалог *Object Properties*.

7. На вкладке **Graphics** ленты, в группе **Active Objects** щелкните **Text**.
8. Щелкните по экрану и введите "Time: #####", затем нажмите Return.
9. Двойной щелчок по объекту откроет диалог свойств объекта *Object Properties*.
10. Щелкните **Text Data Link**.
Диалог *Object Properties* изменится для отображения свойств Text Data Link.
11. В поле **Tag/Expression** введите Time.



Задание тега *Time*

В режиме исполнения система исполнения заменяет символы ##### текстового объекта на значения системного тега *Time*.

12. Закройте диалог *Object Properties*.
 13. Нажмите кнопку приложения (Application button) в левом верхнем углу среды разработки и щелкните **Save** в меню приложения.
- Этот рисунок показывает, как должен выглядеть экран после добавления индикаторов времени и даты.



Индикаторы времени и даты

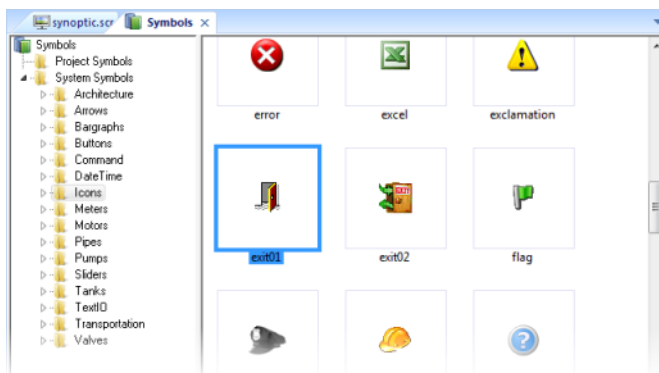
Размещение значка "выход"

В этой части руководства рассказывается, как разместить значок (с использованием Linked Symbol), который позволит пользователю выйти из проекта.

1. На вкладке **Graphics** ленты, в группе **Libraries** щелкните **Symbols**.

Отобразится библиотека символов.

2. В древовидном меню **Symbols** раскройте папку **System Symbols**, затем раскройте подпапку **Icons**.
3. В подпапке **Icons** выберите **exit01**. Символы отобразятся в области просмотра справа от древовидного меню.



Выбор символа "exit01"

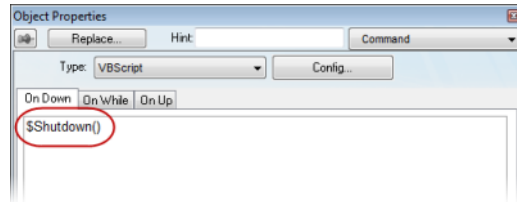
4. Щелкните по символу. Значок курсора мыши изменится, что означает, что символ готов для размещения на экране.
5. Переключитесь назад на экран, на котором вы хотите разместить символ, и щелкните на нем. Символ разместится как объект **Linked Symbol**.



Размещение объекта **Linked Symbol**

6. Пока объект выделен, щелкните **Command** (на вкладке **Graphics** ленты, в группе **Animations**) для применения анимации (animation) к объекту.
7. Двойной щелчок по объекту откроет диалог свойств объекта **Object Properties**.
8. Из списка **Type** выберите **VBScript**.
9. В поле **On Down** (по нажатию) введите `$Shutdown()`.

Shutdown – одна из встроенных в InduSoft Web Studio сценарных функций (scripting functions), которая может быть также использована в макросах VBScript, для этого имя функции предваряется знаком (\$).



Задание выключающей команды для символа

10. Закройте диалог *Object Properties*.

11. Нажмите кнопку приложения (Application button) в левом верхнем углу среды разработки и щелкните **Save** в меню приложения.

Теперь вы можете щелкнуть этот значок в режиме выполнения, произойдет остановка проекта и его завершение.

Тестирование проекта

В этой части руководства рассказывается, как можно протестировать проект.

1. На вкладке **Home** ленты, в группе **Local Management** щелкните **Run**.
Проект запустится, и отобразится стартовый экран.
2. Щелкните на кнопку для открытия экрана мнемосхемы.
Мнемосхема должна отобразиться на экране.
3. Щелкните значок выхода для выключения проекта.

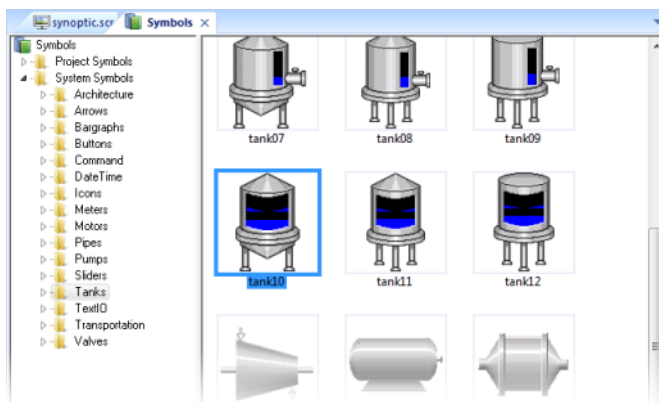
Если какая-нибудь часть проекта работает не так, как нужно, переключитесь назад в среду разработки (по **ALT+TAB**) и щелкните **Stop** на вкладке **Home** ленты.

Размещение анимированной емкости

В этой части руководства рассказывается о том, как выбрать анимированную емкость из библиотеки символов и разместить ее на экране (подобно тому, как это делалось для значка "Exit"), и как связать теги проекта со свойствами емкости.

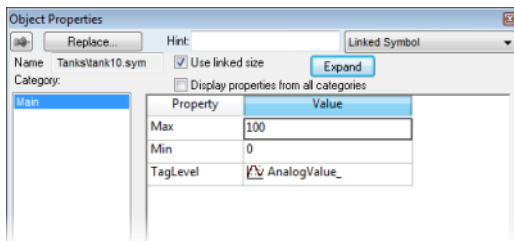
1. На вкладке **Graphics** ленты, в группе **Libraries** щелкните **Symbols**.
2. В древовидном меню **Symbols** раскройте папку **System Symbols**, затем раскройте подпапку **Tanks**.

3. В подпапке Tanks выберите символ емкости.
Вы можете выбрать любую понравившуюся емкость, все они имеют сходные базовый функционал.



Выбор символа емкости

4. Щелкните по символу.
Значок курсора мыши изменится, что означает, что символ готов для размещения на экране.
5. Переключитесь назад на экран, на котором вы хотите разместить символ, и щелкните на нем.
Символ разместится как объект *Linked Symbol*.
6. Двойной щелчок по объекту откроет диалог свойств объекта *Object Properties*.

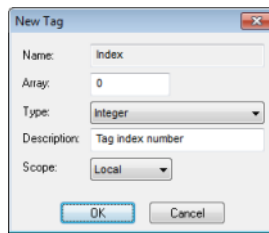


Свойства символа емкости

Емкость представляет собой распределение различных объектов и анимаций (прямоугольники, гистограммы, и т.д.), которые объединены

вместе в связанный символ (Linked Symbol). Вы можете модифицировать свойства этого символа, редактируя список свойств. Для данного проекта потребуется задать связь тега с уровнем жидкости емкости.

7. Для свойства **TagLevel1** удалите текущее значение и впишите `Level[Index]`.
Обратите внимание, что в данном случае не обязательно открывать таблицу **Project Tags** для создания новых тегов.
Так как тег **Index** не был ранее создан. Вместо этого вы получите предупреждающее сообщение в диалоговом окне, с предложением добавить ранее несуществующий тег в базу.
8. Щелкните **Yes**.
Отобразится диалог *New Tag*.
9. Сконфигурируйте новый тег с **Array** = 0, **Type** как **Integer** и **Scope** как **Local**.



Конфигурирование нового тега

10. Щелкните **OK** для закрытия диалога *New Tag*.

Вы можете использовать тег **Index**, чтобы выбрать элемент массива тега **Level1** для отображения уровня выбранной емкости в одном и том же объекте:

- Когда **Index** = 1, объект отображает уровень емкости 1 (т.е. `Level1[1]`);
- Когда **Index** = 2, объект отображает уровень емкости 2 (т.е. `Level1[2]`);
- Когда **Index** = 3, объект отображает уровень емкости 3 (т.е. `Level1[3]`);

Кроме того, так как область видимости тега `local`, тег в одно и то же время может иметь различные значения на сервере и клиентской станции. Следовательно, локальный пользователь (серверная станция) может отслеживать уровень емкости 1, тогда как удаленный пользователь (клиентская станция) может отслеживать уровень емкости 2.

11. Закройте диалог *Object Properties*.
12. Нажмите кнопку приложения (Application button) в левом верхнем углу среды разработки и щелкните **Save** в меню приложения.

Этот рисунок показывает, как должен выглядеть экран после добавления на него объекта емкости.

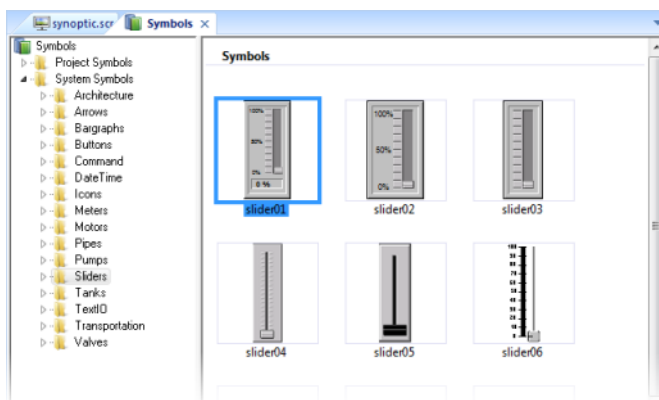


Объект-емкость

Размещение слайдера уровня

В этой части руководства рассказывается, как выбрать слайдер уровня из библиотеки символов и связать его с анимированной емкостью.

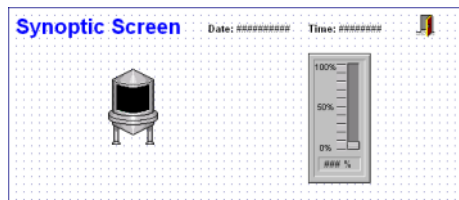
1. На вкладке **Graphics** ленты, в группе **Libraries** щелкните **Symbols**.
2. В древовидном меню **Symbols** откройте папку **System Symbols**, затем откройте подпапку **Sliders**.



Выбор символа слайдера

3. В подпапке **Sliders** выберите слайдер.
Вы можете выбрать любой понравившейся слайдер, все они имеют сходные базовый функционал.
4. Щелкните по символу.
Значок курсора мыши изменится, что означает, что символ готов для размещения на экране.

5. Переключитесь назад на экран, на котором вы хотите разместить символ, и щелкните на нем.
Символ разместится как объект *Linked Symbol*.
 6. Двойной щелчок по объекту откроет диалог свойств объекта *Object Properties*.
 7. Для свойства **TagName** удалите текущее значение и впишите `Level[Index]`.
Аналогично проделанному для символа емкости, вам нужно модифицировать свойство символа, связанное с уровнем слайдера.
 8. Закройте диалог *Object Properties*.
 9. Нажмите кнопку приложения (Application button) в левом верхнем углу среды разработки и щелкните **Save** в меню приложения.
- Этот рисунок показывает, как должен выглядеть экран после добавления на него объекта слайдера.



Объект слайдера уровня

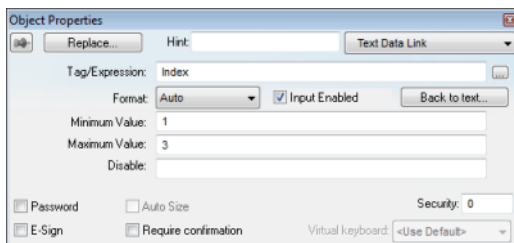
Рисование селектора емкостей

В этой части руководства рассказывается, как нарисовать текстовое поле ввода, которое можно использовать для задания номера емкости, отображаемой на экране.

1. На вкладке **Graphics** ленты, в группе **Active Objects** щелкните **Text**.
2. Щелкните по экрану и введите "Tank: #", затем нажмите Return.
3. Двойной щелчок по объекту откроет диалог свойств объекта *Object Properties*.
4. Щелкните **Text Data Link**.
Диалог *Object Properties* изменится для отображения свойств **Text Data Link**.
5. В поле **Tag/Expression** введите `Index`.
6. Выберите опцию **Input Enabled** (разрешение ввода).

Это позволит оператору вводить новые значения в тег в режиме исполнения.

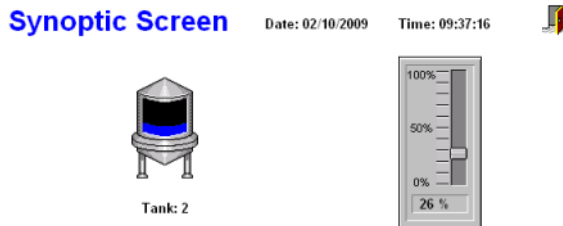
7. В поле **Minimum Value** (минимальное значение) введите 1.
8. В поле **Maximum Value** (максимально значение) введите 3.



Конфигурирование текстового ввода "Tank"

9. Закройте диалог *Object Properties*.
10. Нажмите кнопку приложения (Application button) в левом верхнем углу среды разработки и щелкните **Save** в меню приложения.

Этот рисунок показывает, как должен выглядеть экран после добавления на него селектора емкостей.



Селектор емкости в режиме исполнения

Тестирование проекта

В этой части руководства рассказывается, как можно протестировать проект с анимированной емкостью, слайдером уровня и селектором емкости.

1. На вкладке **Home** ленты, в группе **Local Management** щелкните **Run**. Проект запустится, и отобразится стартовый экран.
2. Щелкните на кнопку для открытия экрана мнемосхемы.

Мнемосхема должна отобразиться на экране.

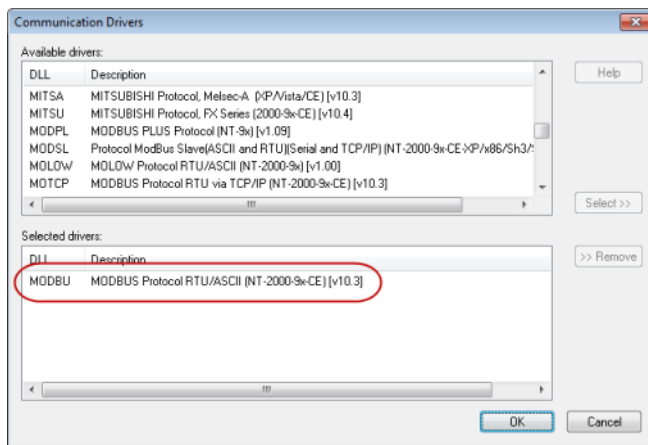
3. Задайте номер емкости (1, 2, 3) с помощью селектора, затем используйте слайдер для установки уровня жидкости в емкости.
Обратите внимание, что установка и отображение уровней каждой емкости происходит независимо от соседних.
4. Щелкните значок выхода для выключения проекта.

Если какая-нибудь часть проекта работает не так как нужно, переключитесь назад в среду разработки (по **ALT+TAB**) и щелкните **Stop** на вкладке Home ленты.

Конфигурирование драйвера связи

В этой части руководства рассказывается, как выбрать и сконфигурировать драйвер для связи с внешним устройством ввода/вывода.

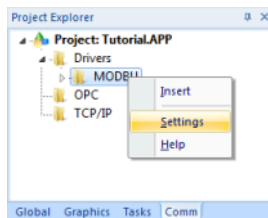
1. В проводнике проекта щелкните по секции **Comm**.
2. Щелкните правой кнопкой мыши по папке **Drivers** и выберите **Add/Remove Drivers** в контекстном меню.
Отобразится диалог *Communication Drivers*.
3. Выберите драйвер из списка доступных драйверов **Available drivers**, затем щелкните **Select**.
Для данного проекта выберите **MODBU**.
Драйвер переместится в список выбранных драйверов **Selected drivers**.



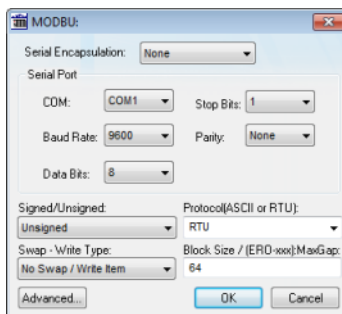
Выбор драйвера **MODBU**

4. Щелкните **OK**.
Диалог *Communication Drivers* закрывается, драйвер добавится в папку драйверов **Drivers** в проводнике проекта.

5. В проводнике проекта щелкните правой кнопкой мыши по папке **MODBU** и выберите **Settings** в контекстном меню.



Отобразится диалог *Communication Settings*.



Диалог *Communication Settings* для драйвера **MODBU**

6. Сконфигурируйте параметры связи необходимые для целевого устройства. Для этого проекта примите параметры по умолчанию.

 **Примечание:** Для получения дополнительной информации о драйвере щелкните **Communication Drivers** на вкладке **Help** ленты.

7. Щелкните **OK** для закрытия диалогового окна.
8. В проводнике проекта щелкните правой кнопкой мыши по папке **MODBU** и выберите **Insert** в контекстном меню. Будет создана и открыта для редактирования новая таблица драйвера с именем `MODBU001.drv`.
9. Сконфигурируйте заголовок таблицы:
 - a) В поле описания **Description** введите `Tutorial Modbus`. Эта настройка нужна только для документирования, она не влияет на выполнение проекта.
 - b) В поле разрешения чтения **Enable Read When Idle** введите `1`.

Этот параметр управляется параметром типа Boolean. Значение 1 либо вводится вручную, как сделано выше, либо является результатом вычисления тега/выражения. Параметр **Enable Read When Idle** определяет режим постоянного опроса внешнего устройства, даже если в данных отсутствуют изменения.

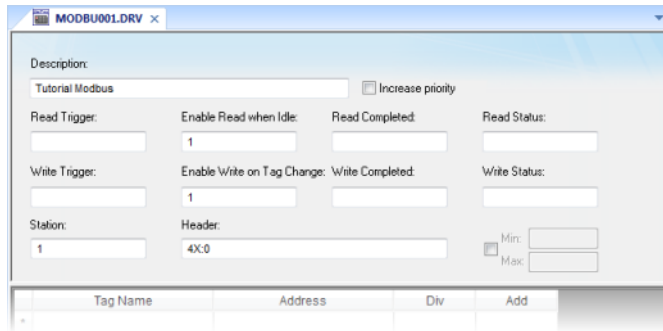
- c) В поле разрешения записи **Enable Write On Tag Change** задайте 1. Этот параметр также управляется величиной Boolean. Значение 1 заставляет проект записывать значение тега во внешнее устройство только при изменении значения тега, а не непрерывно. Это экономит системные ресурсы и повышает производительность в режиме исполнения.
- d) В поле **Station** задайте 1. Значение соответствует номеру (адресу) устройства ввода/вывода, связанного с этим драйвером. Как правило, ПЛК является устройством с адресом 1.
- e) В поле заголовков (пакета) **Header** введите 4X:0.

Вы должны использовать драйвер-ориентированный формат. Для MODBUS это:

```
register_type:initial_offset
```

Тип регистра	Описание
0X	Coil Status
1X	Input Status
3X	Input Register
4X	Holding Register

Тип регистра	Описание
ID	Число Slave ID

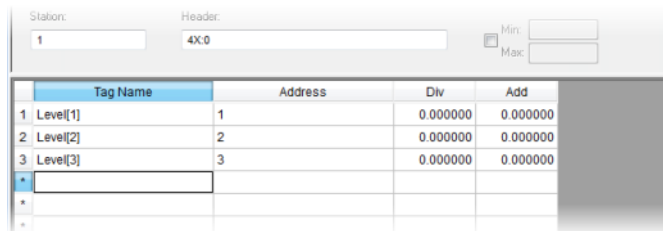


Заполненный заголовок таблицы

10. В теле таблицы введите теги и их соответствующие адреса на устройстве для каждого тега:

- В поле **Tag Name** введите имя тега.
- В поле **Address** введите значение, которое будет прибавляться к смещению из заголовка пакета для формирования полного адреса.

Имя тега	Адрес	Полный адрес устройства
Level [1]	1	4X:1 (Holding Register 1)
Level [2]	2	4X:2 (Holding Register 2)
Level [3]	3	4X:3 (Holding Register 3)



Заполненная таблица

11. Нажмите кнопку приложения (Application button) в левом верхнем углу среды разработки и щелкните **Save** в меню приложения.

12. Когда будет предложено выбрать номер таблицы драйвера, введите 1 и щелкните **OK**.

Отслеживание обмена данными с устройством ввода/вывода в режиме исполнения

В этой части руководства рассказывается, как отслеживать обмен данными с внешним устройством в режиме исполнения с использованием окна журнала *Log*.

1. На вкладке **Home** ленты, в группе **Local Management** щелкните **Run**. Проект запустится, и отобразится стартовый экран.
2. Нажмите **ALT+TAB** для переключения назад в среду разработки.
3. Щелкните правой кнопкой мыши в окне *Output*, затем щелкните **Settings**. Отобразится диалог настроек журнала *Log Settings*.
4. Выберите опции **Field Read Commands**, **Field Write Commands** и **Protocol Analyzer**.
5. Щелкните **OK** для закрытия диалога *Log Settings*.

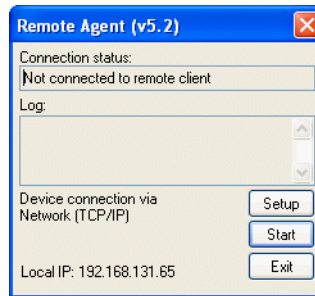
Теперь вы можете отслеживать обмен с внешним устройством в режиме исполнения.

Загрузка проекта на устройство **Windows Embedded**

В этой части руководства рассказывается, как загрузить проект на устройство Windows Embedded, такое как HMI-панель.


После конфигурирования и тестирования проекта (на рабочей станции разработчика), вы можете загрузить его на удаленную станцию с Windows, на которой работает IWS, либо устройство Windows Embedded, на котором работает CEView.

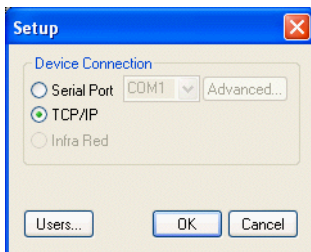
1. На рабочем столе удаленной станции щелкните **Start > All Programs > InduSoft Web Studio v7.0 > Remote Agent**.
Запустится утилита Remote Agent.



Утилита **Remote Agent**

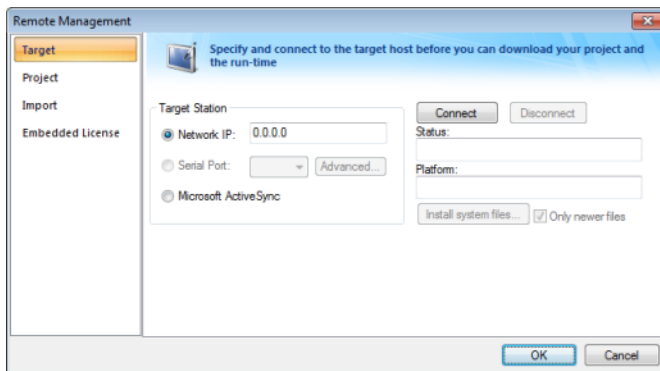
2. Щелкните **Setup**.
Отобразится диалог *Setup*.
3. Выберите тип соединения: **Serial**, **TCP/IP** или **Infrared** – между удаленной станцией и станцией разработчика.

 **Примечание:** Для лучшей производительности мы рекомендуем использовать TCP/IP.



Выбор **TCP/IP** соединения на удаленной станции

4. Щелкните **OK** для закрытия диалога *Setup*, но оставьте работающей утилиту *Remote Agent*.
5. В среде разработки щелкните **Connect** на вкладке **Home** ленты. Отобразится диалог *Remote Management*.



Диалоговое окно **Remote Management**

6. Выберите тип соединения с целевой (удаленной) рабочей станцией. Этот выбор должен соответствовать типу соединения в настройках утилиты *Remote Agent* удаленной станции.
7. Если вы выбрали **Network IP**, задайте IP-адрес удаленной станции.
8. Щелкните **Connect**.

Когда соединение установится, информация об удаленной станции отобразится в полях **Status** и **Platform**.

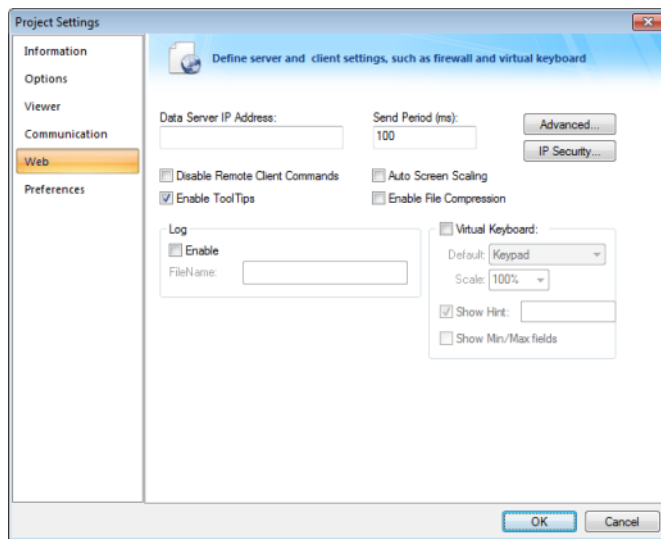
9. Если удаленная станция - устройство Windows Embedded, щелкните **Install system files**.
Системные файлы установятся на удаленную станцию.
10. Щелкните по секции **Project**.
11. Щелкните **Download**.
Проект загрузится на удаленную станцию.
12. Щелкните **Run**.
Проект IWS запустится на удаленной станции.

Развертывание проекта как **web**-приложения

В этой части руководства рассказывается, как развернуть проект в качестве web-приложения, к которому удаленные пользователи смогут получить доступ посредством Internet Explorer.

Для работы через Internet Explorer, в качестве тонкого клиента необходимо установить ActiveX компонент, который "проигрывает" проектные экраны IWS. Если ваш компьютер подключен к Internet, IE автоматически загрузит компонент с публичного сервера InduSoft, когда вы первый раз попытаетесь подключиться к исполняемому проекту.

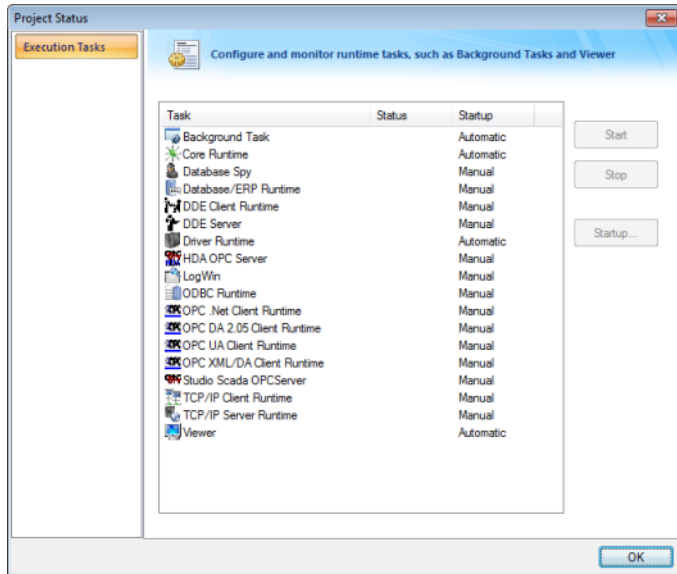
1. Конфигурирование IP-адреса сервера данных (data server).
 - a) На вкладке **Project** ленты, в группе **Web** щелкните **Thin Client**. Отобразится диалог *Project Settings* с выбранной секцией **Web**.



Секция **Web** диалога *Project Settings*

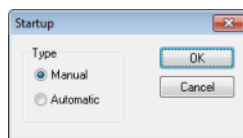
- b) В поле **Data Server IP Address** введите IP-адрес web-сервера. Для данного проекта задайте 127.0.0.1, который является стандартным локальным адресом "localhost".
 - c) Щелкните **OK** для закрытия диалога.
2. Убедитесь, что сервер данных стартует совместно с проектом.

- a) На вкладке **Home** ленты, в группе **Local Management** щелкните **Tasks**.
Отобразится диалог выполняемых задач *Execution Tasks*.



Диалог *Execution Tasks*

- b) В списке задач выберите **TCP/IP Server Runtime** и щелкните **Startup**.
Отобразится диалог *Startup*.



Диалог *Startup*

- c) Выберите **Automatic** и щелкните **OK**.
- d) Щелкните **OK** для закрытия диалога *Execution Tasks*.
3. Сохраните и закройте все открытые экраны и таблицы.
4. Щелкните кнопку приложения в левом верхнем углу среды разработки, выберите **Publish > Save All As HTML** в меню приложения.

Экраны проекта сохраняются как файлы HTML в подпапке Web папки проекта (например, `\project_name\Web`).

5. Конфигурирование web-сервера для открытия доступа по сети к подпапке Web.

Для данного проекта используется NT Web Server - бесплатный, компактный web-сервер, включенный в состав IWS. Просто скопируйте программный файл [...]\`InduSoft Web Studio v7.0\Bin\NTWebServer.exe` в подпапку Web, и запустите его. Сервер автоматически публикует содержимое папки, в которой он находится без дополнительного конфигурирования.



Внимание: NT Web Server предоставлен только для целей тестирования. Не используйте его в реальных промышленных приложениях в защищенных сетях.

Вы также можете сконфигурировать подпапку Web в качестве корневой "root" или домашней "home" директории других web-серверов (например, Microsoft IIS), или можете скопировать содержимое подпапки Web в рабочую директорию web-сервера вашей сети. Web-сервер (который предоставляет файлы HTML клиентам) и сервер данных (который отвечает за обмен данными между клиентами и проектом IWS) могут располагаться на разных компьютерах.

6. На вкладке **Home** ленты, в группе **Local Management** щелкните **Run**.
7. Откройте web-браузер (например, Microsoft Internet Explorer), введите URL-адрес экрана мнемосхемы на web-сервере.
Для данного проекта введите `http://127.0.0.1/synoptic.html`, затем нажмите **Return**. (127.0.0.1 является стандартным локальным IP-адресом "localhost".)
Через некоторое время после того, как браузер загрузит и установит ActiveX компонент, отобразится экран мнемосхемы.

Обратите внимание, что вы можете изменять уровень жидкости в любой емкости с помощью локального интерфейса либо удаленно через web-браузер. И изменения немедленного проявляются на всех клиентах. Они работают одинаково хорошо.