

УТВЕРЖДАЮ

Начальник службы автоматизации

ООО «НПП ОЗНА-Инжиниринг»

\_\_\_\_\_\_\_\_\_\_В.О. Литвинов

«\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_ 20\_\_ г.

**Система измерений количества и**

**показателей качества нефти**

**№102 ПСП «Тайшет-2»**

**ООО «Транснефть - Восток»**

ОИ 410-01.00.00.00.000 И33.4

**Руководство программиста верхнего уровня.**

**Руководство программиста. Часть 4**

РАЗРАБОТАЛ

Главный специалист по РПО МСО

ООО «НПП ОЗНА-Инжиниринг»

\_\_\_\_\_\_\_\_\_\_\_А.С. Русских

«\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_ 20\_\_ г.

Уфа 2020

# СОДЕРЖАНИЕ

|  |
| --- |
| Подпись и дата |

|  |
| --- |
| Инв. № дубл. |

|  |
| --- |
| Взам. инв. № |

|  |
| --- |
| Подпись и дата |

|  |
| --- |
| Инв. № подл. |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

Изм.

Лист

№ докум.

Подпись

Дата

Лист

2

**ОИ 410-01.00.00.00.00 И33.4**

Разраб.

А.С. Русских

Провер.

А.А. Решетников

Н. Контр.

*М.В. Самкова*

Утверд.

В.О. Литвинов

Руководство программиста

верхнего уровня

Лит.

Листов

51

**ООО «НПП ОЗНА-Инжиниринг»**

[СОДЕРЖАНИЕ 2](#_Toc469908130)

[1 ВВЕДЕНИЕ 3](#_Toc469908131)

[2 ОБЩИЕ СВЕДЕНИЯ 4](#_Toc469908132)

[3 ВВЕДЕНИЕ В СИСТЕМУ ОТОБРАЖЕНИЯ ВЕРХНЕГО УРОВНЯ 5](#_Toc469908133)

[3.1 Общие сведения 5](#_Toc469908134)

[3.2 Application Manager (Менеджер приложений) 6](#_Toc469908135)

[3.3 WindowMaker (Режим разработки) 7](#_Toc469908136)

[3.3.1 Общие сведения 7](#_Toc469908137)

[3.3.2 Использование цветовой палитры 9](#_Toc469908138)

[3.3.3 «Горячие кнопки» мыши 10](#_Toc469908139)

[3.3.4 Создание окон приложения 12](#_Toc469908140)

[3.3.5 Открытие, сохранение и закрытие окон 13](#_Toc469908141)

[3.3.6 Объекты WindowMaker 14](#_Toc469908142)

[3.3.7 Анимация ввода данных 20](#_Toc469908143)

[3.3.8 Скрипты для Windows элементов управления 25](#_Toc469908144)

[3.4 Серверы ввода/выводы в InTouch 27](#_Toc469908145)

[3.4.1 Поддерживаемые коммуникационные протоколы 27](#_Toc469908146)

[3.4.2 Особенности адресации 30](#_Toc469908147)

[3.4.3 Обмен данными с другими приложениями 32](#_Toc469908148)

[3.4.4 Определение имени доступа в словаре переменных 33](#_Toc469908149)

[3.5 Тренды 38](#_Toc469908150)

[3.5.1 Архивирование значений переменной 38](#_Toc469908151)

[3.5.2 Отображение 40](#_Toc469908152)

[3.5.3 Изменение параметров в режиме исполнения 43](#_Toc469908153)

[3.5.4 Система распределенных архивов 44](#_Toc469908154)

[3.6 Встроенные языки программирования 45](#_Toc469908155)

[3.6.1 Скрипты 46](#_Toc469908156)

[3.6.2 Встроенные функции 47](#_Toc469908157)

[3.6.3 Функции Quick Functions 48](#_Toc469908158)

[ПРИЛОЖЕНИЕ А 50](#_Toc469908159)

[Лист регистрации изменений 51](#_Toc469908160)

# ВВЕДЕНИЕ

Настоящее руководство содержит описание и сведения, необходимые для понимания принципов построения и корректировки программного обеспечения автоматизированного рабочего места (АРМ) оператора, входящего в состав СОИ СИКН.

Пользователь, осуществляющий процесс корректировки ПО должен обладать необходимым уровнем знаний и навыков для работы с прикладным программным обеспечением, иметь общее представление о системе в целом и о работе на персональной электронно-вычислительной машине (ПЭВМ), уметь пользоваться настоящим руководством и инструкциями служб предприятия.

# ОБЩИЕ СВЕДЕНИЯ

СОИ СИКН предназначена для автоматизированных измерений, сбора, обработки, отображения и регистрации показателей качества нефти.

Система обеспечивает централизованный контроль состояния объекта, сигнализацию отклонения параметров от нормы, дистанционное управление работой объекта, формирование журнала аварийных и технологических сообщений, формирование и печать отчетных документов, ведение базы данных.

Автоматизированное рабочее место (АРМ) предназначено для:

1. наглядного представления в графическом виде состояния технологического процесса и исполнительных механизмов;
2. оповещения оператора при отклонении технологического процесса от заданного режима работы или диагностируемой неисправности оборудования;
3. дистанционного управления исполнительными механизмами;
4. настройки программного обеспечения системы.

АРМ оператора представляет собой полноэкранную систему мониторинга и управления объектами СОИ СИКН. Основным элементом верхнего уровня является SCADA-система Intouch производства Wonderware. Информационный обмен с ИВК осуществляется посредством драйвера KEPServerEx, реализующего интерфейс протокола Modbus TCP. Информационный обмен с ПЛК осуществляется посредством драйвера SimaticNet, реализующего интерфейс PROFINET протокола Siemens S7.

# ВВЕДЕНИЕ В СИСТЕМУ ОТОБРАЖЕНИЯ ВЕРХНЕГО УРОВНЯ

## Общие сведения

Wonderware InTouch– это открытый и расширяемый человекомашинный интерфейс с передовыми графическими возможностями, мощными средствами разработки и гибкой архитектурой для быстрого создания и развертывания приложений автоматизации, позволяющий пользователям связываться с практически любыми промышленными устройствами и системами.

Основные функции Wonderware Intouch:

а. сбор данных;

б. прямое взаимодействие с устройствами ввода/вывода на предприятии;

в. интерфейс к контроллерам при помощи драйверов ввода/вывода;

г. управление данными;

д. обработка собранных данных;

е. мониторинг процесса (графическое отображение);

ж. диспетчерское управление;

з. тревоги, отчеты, архивирование.

Wonderware InTouch позволяет пользователям быстро создавать и развертывать графические представления промышленных процессов реального времени.

InTouch предлагает широкий спектр примитивов для построения графических символов, включая дуги, хорды, секторы, кривые и стандартные элементы управления Microsoft Windows, а также инструменты для моделирования сложных форм путем комбинирования более простых.

Расширенные возможности манипулирования графическими атрибутами обеспечивают эффективное и точное управление посредством встроенных инструментов анимации и разработки сценариев. В инструментарий графических средств включены возможности изменения прозрачности, затушевывания с цветовыми переходами, стили заливки и линий, изменения ориентации, размера, позиции и другое, что позволяет практично и эффективно отобразить автоматизируемые объекты.

Для построения и запуска приложений InTouch, используется три компонента:

* Application Manager (Менеджер приложений) для управления имеющимися приложениями;
* WindowMaker для создания HMI приложений;
* WindowViewer для исполнения HMI приложений.

## Application Manager (Менеджер приложений)

Application Manager включает в себя утилиты для управления InTouch приложениями. Application Manager используется для создания новых приложений или открытия существующих приложений в WindowMaker или WindowViewer.

Application Manager

WindowMaker

WindowViewer

Stand-alone InTouch Applications

Start

Start

Manage

Manage

Create

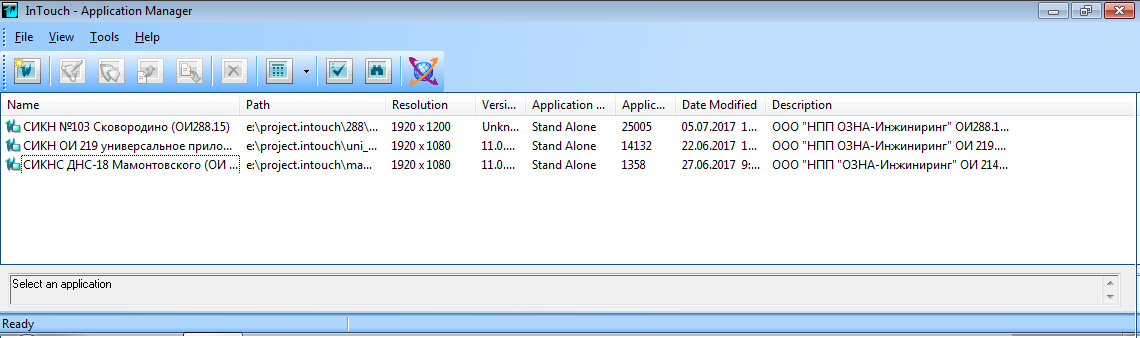
Run

В Application Manager в списке отображается текущие приложения InTouch. Для каждого приложения отображается разрешения экрана, версия и дата изменения.

Application Manager используется также для:

* Поменять свойства приложения InTouch, например имя или описание;
* Осуществлять импорт и экспорт данных из/в базы данных тегов приложения InTouch при помощи утилит DBDump и DBLoad;
* Сконфигурировать WindowViewer как службу.

После открытия Application Manager (Пуск > Wonderware > InTouch), в окне отобразится список приложения InTouch.



В Application Manager отображаются приложения InTouch со следующей информацией:

* Имя приложения;
* Путь приложения;
* Версия приложения;
* Версия InTouch;
* Описание приложения;
* Разрешение экрана.

## WindowMaker (Режим разработки)

Для разработки приложений используется среда WindowMaker. Использовать инструменты объектно-ориентированной графики для создания анимированных окон и окон сенсорных дисплеев. Данные окна могут быть подключены к промышленным системам или другим приложениям MicrosoftWindows.

В WindowMaker есть утилиты и инструменты для создания приложения InTouch:

* Базовые элементы – прямоугольники, окружности, линии и изображения, к которым можно привязать анимацию и реакцию на действие пользователя.
* Элементы управления, которые отображают данные или алармы.
* База данных предварительно созданных промышленных символов и графических элементов.
* Скриптовый язык и набор встроенных функций для управления данными.

### Общие сведения

Используя диалоговое окно свойств WindowMaker Properties, можно сконфигурировать параметры и опции, влияющие на WindowMaker:

• Изменить текст заголовка.

• Отображать сетку или выключить отображение сетки.

• Изменить расстояние между точками сетками.

• Отобразить счетчик тегов.

• Поменять стандартные шрифты текста и кнопок. Установить точность для выбора линии. Установить опцию на закрытие WindowMaker при переключении в WindowViewer.

• Разрешить кнопку быстрого переключения между WindowMaker и WindowViewer.

• Установить число уровней отмены действий.

Для установки свойств WindowMaker

1. В меню Special, выбрать опцию Configure и нажать WindowMaker. Появиться окно свойств WindowMaker.

2. В поле WindowMaker Title Bar, сконфигурировать отображение заголовка. Можно сделать следующее:

• В поле Title Bar, ввести текст, который будет появляться в заголовке в режиме разработки.

• Выбрать опцию Show Application Directory, для включения пути папки приложения в заголовок.

3. В разделе Grid Coordinates, конфигурируется параметры сетки. Можно сделать следующее:

• В поле Spacing, ввести количество пикселов между точками сетки.

• Выбрать опцию Show Grid для отображения сетки.

4. Конфигурирование различных свойств окна. Можно сделать следующее:

• Выбрать опцию Show Tag Count для отображения в меню количества тегов зарегистрированных в словаре. Это очень полезная опция, если производится разработка проекта с ограниченным количеством тегов. Счетчик тегов не включает в себя ссылки на удаленные теги. Для выявления использования ссылок на удаленные теги, необходимо в меню Special выбрать опцию Close on Transfer to WindowViewer, для автоматического закрытия WindowMaker при запуске WindowViewer. Данная опция предназначена для оптимизации работы и экономии ограниченных ресурсов оперативной памяти. Если ограничений нет, и различия по производительности не заметны, то отмечать данную опцию не следует. Когда выбрана опция Close on Transfer to WindowViewer, то автоматически выбирается соответствующая опция Close WindowViewer (в окне свойств WindowViewer Properties).

• Выбрать опцию Pick Through Hollow Objects, для возможности выбора объектов, которые находятся за выделенным объектом. Данная опция дает возможность, например, выбирать объект внутри рамки без того чтобы отсылать рамку на задний фон.

• Выбрать опцию Enable Fast Switch, для использования кнопки “быстрого переключения” между WindowMaker и WindowViewer. При выборе данной опции, в WindowMaker в правом верхнем углу возникнет кнопка Runtime. А в WindowViewer – возникнет кнопка Development. При использовании кнопки быстрого переключения WindowMaker автоматически сохраняет все изменения, произведенные со всеми открытыми окнами, перед переключением в WindowViewer.

• В поле Line Selection Precision, вводится количество пикселов, насколько курсор может отступать от линии, все еще имея возможность ее выбрать. В большинстве случаев, оставляется значение по умолчанию 4 пикселя.

• В поле Levels of Undo, вводится количество уровней отмены или повторения действий. Можно определить максимум до 25 уровней. Если ввести 0, то функция отмены/повтора действия будет выключена. Один уровень – это одно действие. Функции отмены и повтора действий не содержат информации при создании нового окна или открытии окна. Функции также очищаются при закрытии окна.

5. В разделе Configure Fonts, нажать Text или Button, для установки шрифта для текста и кнопок. Выбрать шрифт и нажать ОК. Можно в любом окне изменить данную настройку при помощи панели инструментов Font (шрифт).

6. Нажать ОК.

7. Перезапустить WindowMaker для применения изменений.

### Использование цветовой палитры

Можно использовать цветовую палитру для статических и динамических свойств линий, прямоугольников, скругленных прямоугольников, эллипсов, многоугольников и текста. Можно выбрать цвет фона окон и цвет прозрачности для изображений, который позволяет видеть объекты, находящиеся за изображением. Палитра предлагает широкий выбор цветов, до 16,7 миллионов цветов. Доступные цвета могут быть ограничены возможностями видеокарты.

Так же можно:

• Определять и добавлять различные цвета.

• Импортировать цветовые палитры созданные в других приложениях Windows.

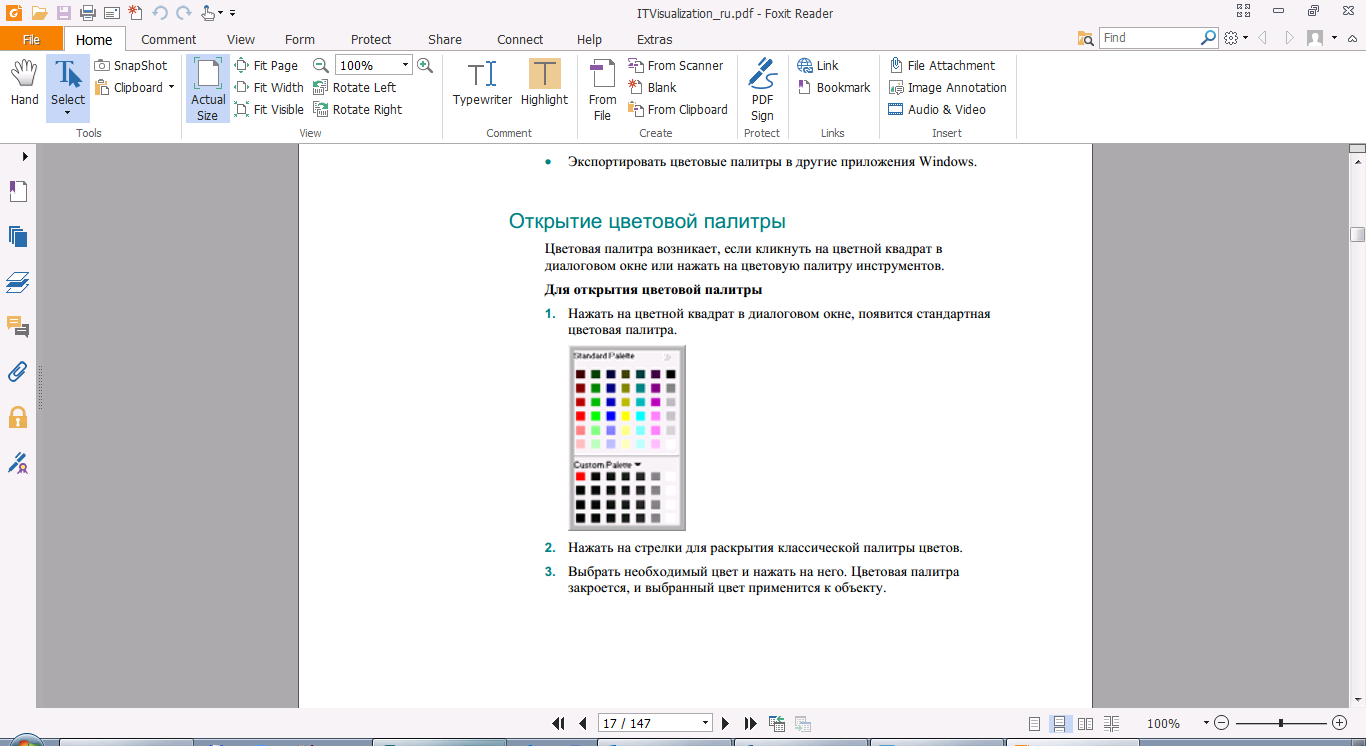
• Экспортировать цветовые палитры в другие приложения Windows.

Цветовая палитра возникает, если кликнуть на цветной квадрат в диалоговом окне или нажать на цветовую палитру инструментов.

Для открытия цветовой палитры

1. Нажать на цветной квадрат в диалоговом окне, появится стандартная

цветовая палитра.



2. Нажать на стрелки для раскрытия классической палитры цветов.

3. Выбрать необходимый цвет и нажать на него. Цветовая палитра закроется, и выбранный цвет применится к объекту.

### «Горячие кнопки» мыши

Использовать следующие “горячие кнопки”, для открытия диалоговых окон и других общих задач.

*Для доступа к командам меню для элементов WindowMaker:*

Нажать правой кнопки мыши на элементе. Элементы включают в себя – графические объекты, папки и т.д.

*Для открытия диалогового окна анимационных связей:*

Двойное нажатие левой кнопкой мыши на объекте или символе.

*Для открытия диалогового окна выбора тега (Tag Browser)*

Двойное нажатие левой кнопкой мыши в поле Expression, в диалоговом окне привязки анимационной связи. Появится окно Select Tag.

*Для доступа к полям тегов:*

В любом поле ввода тега или выражения, ввести имя тега и точку, и потом два раза нажать левую кнопку мыши справа от точки. Откроется окно выбора поля – Choose Field name.

*Для открытия определения тега в словаре тегов*

Двойное нажатие левой кнопкой мыши на теге.

*Перемещение объектов кнопками перемещения курсора*

В WindowMaker, можно использовать кнопки перемещения курсора (кнопки со стрелами), для перемещения выделенного объекта или групп объектов. При перемещении объектов кнопками, то насколько далеко будет перемещен объект, будет зависеть от того, включена сетка или нет. Когда сетка включена, то насколько пикселов переместится объект,будет зависеть то расстояния между точками сетки, которое устанавливается в диалоговом окне свойств WindowMaker.

Стандартно между точками сетки 10 пикселов.

Когда сетка отображается:

• Нажатие на кнопку перемещения курсора, переместит объект на один шаг сетки.

• Нажатие на кнопку SHIFT + кнопку перемещения курсора, переместит объект на два шага сетки.

• Нажатие на кнопку CTRL + кнопку перемещения курсора, переместит объект на четыре шага сетки.

Когда сетка не отображается:

• Нажатие на кнопку перемещения курсора, переместит объект на один пиксель.

• Нажатие на кнопку SHIFT + кнопку перемещения курсора, переместит объект на десять пикселов.

• Нажатие на кнопку CTRL + кнопку перемещения курсора, переместит объект 50 пикселов.

Когда сетка выключена:

*Для перемещения объекта или группы объектов на один пиксель*

Нажать кнопку перемещения курсора.

*Для перемещения объекта или группы объектов на 10 пикселов*

Удерживая SHIFT нажать кнопку перемещения курсора.

*Для перемещения объекта или группы объектов на 50 пикселов*

Удерживая CTRL нажать кнопку перемещения курсора

### Создание окон приложения

При создании нового окна приложения, стандартные настройки будут автоматически подставляться из ранее созданного окна или открытого в данный момент окна. Имя окна может содержать не более 32 символов и может включать любые символы клавиатуры кроме кавычек. При создании окна необходимо указать только имя название окна, остальные параметры являются опциональными. Можно добавить комментарий для окна, но это может использоваться только в информационных целях. Комментарий окна не используется приложением. Стандартно, размеры окна установлены равными размерам ранее созданного окна. Эти размеры также автоматически изменяются, если их изменить вручную, перетаскиваем границ окна. Если создать приложение в системе с классической темой Windows, а затем запустить приложение на системе с темой WindowsХР, то окна InTouch могут открываться обрезанными в нижней части на несколько пикселов. Это происходит потому, что заголовок окна в классической теме Windows уже чем в теме WindowsХР.

Для создания нового окна

1. В меню File, выбрать New Window. Появится диалоговое окно свойств окна.

2. Для конфигурирования основных свойств окна, необходимо сделать следующее:

• В поле Name ввести имя окна, которое будет отображаться в заголовке.

• В поле Comment, ввести комментарий для данного окна.

• В поле Window Color выбрать фоновый цвет окна.

3. В поле Window Type, можно сконфигурировать как будет открываться окно в режиме исполнения.

• Replace – это тип окна, который при открытии автоматически закрывает любые пересекающиеся с ним окна.

• Overlay – тип окна, который при открытии возникает поверх всех открытых окон. Это окно может быть больше чем окно(а), которые оно перекрывает. Когда перекрывающее окно закрывается, все окна за ним появляются снова. Нажатие на любую точку окна находящегося за перекрывающим окном, вызовет перемещение этого окна на передний план (и оно станет активным).

• Popup – это типа окна, которое всегда будет располагаться поверх всех окон. Окнам данного типа обычно требуется действие пользователя для закрытия.

4. В разделе Frame Style, можно сконфигурировать границу вокруг окна.

• Single – окно с объемной границей, которое может иметь заголовок и возможно изменения размеров окна. Выбрать опцию Title Bar для отображения заголовка.

• Double - окно с объемной границей, которое не может иметь заголовок и не возможно изменение размеров без опции Size Controls.

5. Выбрать опцию Size Controls, для того чтобы разрешить пользователям изменять размеры окна в режиме исполнения.

6. В разделе Dimensions, можно определить положение и размеры окна:

• В поле X location – ввести число пикселей между левым краем области разработки и левым краем определяемого окна.

• В поле Y location – ввести число пикселей между верхним краем области разработки и верхним краем определяемого окна.

• В полях Window Width и Window Height ввести ширину и высоту окна в пикселях.

7. Нажать ОК.

### Открытие, сохранение и закрытие окон

При разработке приложения, можно открыть окон столько сколько позволит производительность компьютера.

Для открытия окна

1. В меню File, выбрать Open Window. Откроется диалоговое окно Windows to Open, со списком имен всех окон приложения.

2. Сделать следующее:

• Чтобы открыть одно окно – два раза нажать на имя окна.

• Чтобы открыть несколько окон, выбрать из списка несколько окон и нажать ОК. При сохранении окна, все графические объекты, скрипты, свойства, задействованные в приложении, также будут сохранены.

Для сохранения окна

1. В меню File, выбрать Save Window. Откроется диалоговое окно Windows to Save, со списком имен всех окон приложения.

2. Выбрать окна, которые необходимо сохранить.

3. Нажать ОК.

При закрытии окон, которые были изменены, появится сообщение, предлагающее сохранить изменения. Для закрытия окна

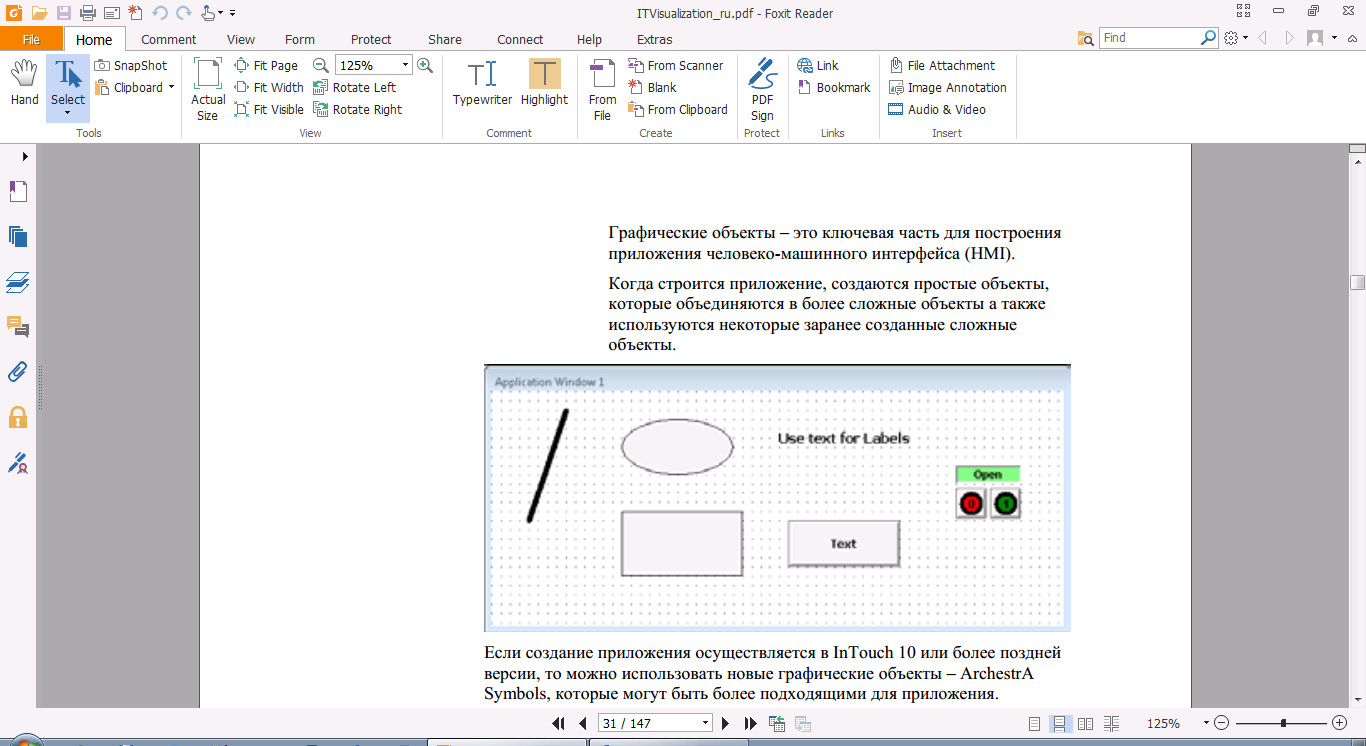
4. В меню File, выбрать Close Window. Откроется диалоговое окно Windows to Close, со списком имен открытых окон приложения.

5. Выбрать окна, которые необходимо закрыть.

6. Нажать ОК.

### Объекты WindowMaker

Графические объекты – это ключевая часть для построения приложения человеко-машинного интерфейса (HMI). Когда строится приложение, создаются простые объекты, которые объединяются в более сложные объекты а также используются некоторые заранее созданные сложные объекты.



#### Простые объекты

Можно создавать следующие типы простых объектов:

• Линии

• Фигуры

• Текст

• Кнопки

Каждый простой объект имеет атрибуты, которые управляют им

• Цвет и ширина линии.

• Цвет заполнения.

• Ширина

• Высота

• Ориентация.

#### Создание линий и фигур

Кнопки рисования расположены в панели инструментов рисования.

|  |  |
| --- | --- |
| Объект | Нажать |
| Линию | Кнопку с линией |
| Горизонтальную или вертикальную линию | Кнопка с пересечением линий |
| Прямоугольник | Кнопка я прямоугольником |
| Прямоугольник с закругленными краями | Кнопка я прямоугольником с закругленными краями |
| Круг или эллипс | Кнопка с эллипсом. Нажать и удерживая SHIFT, нарисовать круг. |

#### Создание Кнопок

Для создания точек взаимодействия с приложением можно использовать кнопки. Процесс создания кнопки сходен с созданием графических объектов. Для более подробной информации по созданию многоугольников, см. ниже раздел Создание многоугольников и ломаных линий. Для создания кнопки

1. В панели инструментов рисования нажать кнопку Button.

2. Нажать ее, а затем нажать в окно приложения и задать размеры кнопки.

3. Изменить стандартный текст кнопки.

a. Нажать правую кнопку мыши и в контекстном меню выбрать Substitute Strings.

b. В поле New String, ввести новый текст для кнопки.

c. Нажать ОК.

#### Создание многоугольников и ломаных линий

Создание ломаных линий немного отличается от создания простых линий.

Для создания многоугольников и ломаных линий

1. В панели инструментов рисования нажать кнопку Polyline или Polygon.

2. Нажать в окне приложения для создания начальной точки.

3. Далее аналогичным образом создать остальные точки, для определения многоугольник или ломаную линию.

4. На последней точке нажать два раза.

#### Создание текста

В приложении можно использовать так же и текстовые объекты. При создании текста, опции установки форматирования текста будут, использованы из свойств WindowMaker. Можно изменить внешний вид выделенного текста. Когда вводится текст в виде нескольких строк, они становятся объектами, которые могут быть перемещены и редактированы независимо. Можно также комбинировать текстовые объекты в символы и редактировать их как группу.

#### Для создания текстового объекта

1. В панели инструментов рисования нажать кнопку Text.

2. Нажать в окне приложения для создания точки начала текста.

3. Ввести текст и нажать ENTER

#### Общие действия

Нажатием правой кнопки мыши на объекты откроется меню, отображающее какие действия можно совершить с объектом. Можно:

• Выделение объектов

• Выравнивание объектов

• Наложение объектов

• Разворачивание объектов

• Изменение размера объектов

• Изменение шрифта

• Изменение цвета заполнения

• Управление расположением по вертикали и горизонтали.

• Перемещение объектов

• Упорядочивание объектов

• Отменять изменения

• Разворачивание символов

• Вращение объектов

• Изменение линий или контуров

• Удаление объектов

*Выделение объектов*

Перед тем как изменять объект, необходимо его выделить. При выделении объекта, вокруг него по периметру образуется рамка. Можно использовать ее для изменения размеров объекта.

Для выделения всех объектов в активном окне

В меню Edit, выбрать Select All, или нажать F2.

Для выделения объекта - нажать кнопку Select Mode , а затем нажать на объект который необходимо выделить.

Для отмены выделения объекта - Нажать на любую точку в пустой области окна.

Для выделения нескольких объектов - нажать кнопку Select Mode, выбрать первый объект, а затем, удерживая SHIFT нажимать на остальные объекты.

Для выделения группы объектов - нажать кнопку Select Mode, а затем мышкой нарисовать квадрат вокруг объектов. Все объекты внутри прямоугольника будут выделены.

Для отмены выделения определенного объекта или объектов из группы удерживая SHIFT + нажать на объект.

*Перемещение объектов*

Перемещать объекты можно:

• При помощи кнопок перемещения курсора на клавиатуре

• Ввести координаты относительно окна в полях в строке состояния.

При перемещении объекта, координаты в строке состояния изменяются.

*Перетаскивание объекта*

Выделить объект и перетащить его. При перемещении объект кнопками курсора, то насколько далеко он будет перемещен, будет зависеть от того, отображается сетка или нет. При включенной сетке, объект перемещается на то расстояние в пикселях, которое указано в свойствах WindowMaker – расстояние между точками сетки. Стандартно расстояние между точками сетки 10 пикселов.

Когда сетка отображается

• Нажатие кнопки курсора, переместит объект на одно деление сетки.

• Удерживая SHIFT + кнопку курсора, переместит объект два деления сетки.

• Удерживая CTRL + кнопку курсора, переместит объект четыре деления сетки.

Когда сетка не отображается

• Нажатие кнопки курсора, переместит объект на один пиксель.

• Удерживая SHIFT + кнопку курсора, переместит объект на 10 пикселов.

• Удерживая CTRL + кнопку курсора, переместит объект на 50 пикселов.

Для перемещения объекта кнопками курсора

Выделить объект

• Нажать кнопку курсора

• SHIFT + кнопку курсора

• CTRL + кнопку курсора

Для перемещения объекта из строки состояния

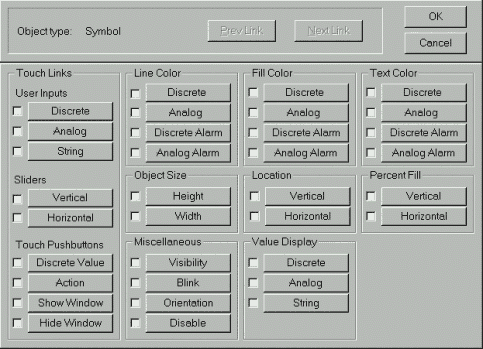
1. Выделить объект

2. Ввести X и Y координаты в строке состояния.

3. Нажать ENTER.

#### Динамические свойства

Один и тот же объект может иметь набор различных динамических свойств. Комбинации этих свойств предоставляют возможность создавать на экране в режиме исполнения (Runtime) практически любые динамические эффекты. Для установки динамических свойств надо прежде всего вызвать на экран диалог их выбора (рисунок 3.3.6.8 - 1). Это достигается командой Special/Animation Link или двойным щелчком левой кнопки мыши на объекте.



*Рисунок 3.3.6.8 - 1 Диалог выбора динамических свойств объекта.*

Все динамические связи можно разделить на две группы: Touch Links (левая колонка) и Display Links (три колонки справа). С помощью свойств Touch Links выполняется какой - либо ввод в систему. Свойства Display Links осуществляют вывод информации на экран дисплея.

Нажатие на любую клавишу диалога (рисунок 3.3.6.8 - 1) вызывает появление нового диалога для определения соответствующего свойства объекта. Количество диалогов соответствует количеству динамических свойств (кнопок) диалога выбора. Все диалоги различны, но большинство из них имеет общие характеристики:

- окно типа объекта;

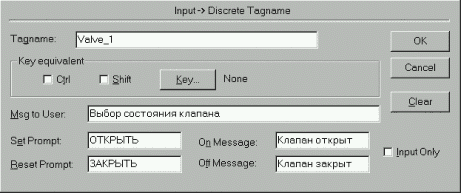
- одинаковую палитру цветов;

- быстрый вызов словаря переменных;

- быстрый доступ к полям переменных;

- поддержку правой кнопки мыши в полях Tagname (имя переменной) и Expression (выражение).

На рисунок 3.3.6.8 – 2 приведен диалог для определения свойств объекта (кнопки), управляющего значением дискретной переменной.



*Рисунок 3.3.6.8 - 2 Диалог определения свойств кнопки.*

Завершение работы с диалогом производится нажатием кнопки Ok. Если переменная поля Tagname была ранее определена в словаре переменных данного приложения, пользователь возвращается в диалог выбора динамических свойств объекта (рисунок 3.3.6.8 – 1). Можно либо продолжить определение других динамических свойств для данного объекта, либо, нажав Ok, вернуться на поле разработки окна приложения.

### Анимация ввода данных

Для создания объектов, позволяющих оператору взаимодействовать с системой, необходимо использовать анимационные связи на нажатие. Например, оператор может войти в систему при помощи ввода имени пользователя и пароля с клавиатуры, открыть или закрыть клапан, ввести новую уставку аларма, запустить или остановить процесс и так далее. Вокруг реагирующих на нажатие объектов, появляется рамка, когда они оказывается в фокусе т.е курсор мыши перемещается поверх объекта, или нажимается кнопка TAB или кнопки курсора и т.д. Если предполагается что пользователь будет кнопкой TAB переходить от одного реагирующего на нажатие объекта к другому, то необходимо расположить их по горизонтали. Нажимая кнопку TAB, фокус перемещается от одного объекта к другому слева направо, с верху вниз окна. Оператор может активировать объект, реагирующий на нажатие, нажав на не него мышкой, или нажав привязанную к нему кнопку, или нажать Enter после появления рамки вокруг объекта, или просто нажать на него, если используются устройства с сенсорным экраном. Можно определить девять типов анимационных связей:

|  |  |
| --- | --- |
| Тип анимационной связи | Действие |
| User Inputs | • Discrete  • Analog  • String |
| Sliders | • Vertical  • Horizontal |
| Pushbuttons | • Action  • Show Window  • Hide Window |

Когда текстовой поле используется для ввода пользователем информации, то текст появляется на экране после нажатия кнопки. Если не Вы не хотите чтобы текст появлялся на экране по мере ввода, то необходимо при конфигурации выбрать опцию Input Only.

*Дискретный ввод данных*

Можно разработать объект с анимационной связью для изменения оператором значений дискретных тегов. Например, включение или выключение насоса.

Для создания анимационной связи дискретного ввода данных:

1. Нажать на объекте правой кнопкой мыши и в контекстном меню выбрать Animation Links. Появится соответствующее диалоговое окно.

2. В разделе Touch Links, нажать User Inputs и выбрать Discrete. Появится окно Input-> Discrete Input.

3. В поле Tagname, ввести имя дискретного тега или выражение, результат которого будет принимать дискретное значение.

4. Дополнительно можно назначить эквивалентную кнопку на клавиатуре в разделе Key Equivalent.

5. Для конфигурирования опций, сделать следующее:

• В поле Msg to User ввести сообщение, которое будет возникать в поле ввода.

• В поле Set Prompt и Reset Prompt, ввести сообщение, которое будет появляться на кнопках, которые оператор будет нажимать.

• В поле On Message и Off Message ввести сообщения, которые будут отображаться в текстовых полях, привязанных к объекту.

6. Выбрать Input Only, для того чтобы текст не отображался в текстовом

поле, привязанном к объекту.

7. Нажать ОК.

Аналоговый ввод данных

Можно разработать объект с анимационной связью для изменения оператором значений аналоговых тегов. Например, уставки алармов, скорость конвейера

Для создания анимационной связи аналогового ввода данных:

1. Нажать на объекте правой кнопкой мыши и в контекстном меню выбрать Animation Links. Появится соответствующее диалоговое окно.

2. В разделе Touch Links, нажать User Inputs и выбрать Analog. Появится окно Input-> Analog Input.

3. В поле Tagname, ввести имя аналогового тега или выражение

4. Дополнительно можно назначить эквивалентную кнопку на клавиатуре в разделе Key Equivalent.

5. Для конфигурирования опций, сделать следующее:

• В разделе Keypad?, выбрать Yes если необходимо отображать на экране цифровую клавиатуру для ввода нового значения. В разделе Msg to User, ввести сообщение, которое будет возникать на клавиатуре.

• В полях Min Value и Max Value ввести минимальное и максимальное значения для вводимого значения тега..

6. Выбрать Input Only, для того чтобы текст не отображался в текстовом поле, привязанном к объекту.

7. Нажать ОК.

*Текстовый ввод данных*

Можно разработать объект с анимационной связью для ввода оператором текста. Например, имени партии, идентификатора оператора, пароля.

Для создания анимационной связи текстового ввода данных:

1. Нажать на объекте правой кнопкой мыши и в контекстном меню выбрать Animation Links. Появится соответствующее диалоговое окно.

2. В разделе Touch Links, нажать User Inputs и выбрать String. Появится окно Input-> String Input.

3. В поле Tagname, ввести имя строкового тега или выражение, результат которого будет строковым.

4. Для конфигурирования опций, сделать следующее:

• Дополнительно можно назначить эквивалентную кнопку на клавиатуре в разделе Key Equivalent.

• В разделе Keypad?, выбрать Yes если необходимо отображать на экране цифровую клавиатуру для ввода нового значения. В разделе Msg to User, ввести сообщение, которое будет возникать на клавиатуре.

5. Выбрать Echo Characters?, для того чтобы текст выбрать отображался ли вводимый текст в поле ввода.

• Нажать Yes для отображения текста в поле ввода.

• Нажать No для запрета отображения текста в поле ввода.

• Нажать Password, для “маскирования” символов вводимого текста. В поле Password Char, выбрать символ маскирования. Выбрать Encrypt, для шифрования пароля.

Важно Шифрование пароля работает только в контексте InTouch HMI.Не надо шифровать строку, если необходимо передать ее во внешнюю систему безопасности, такую как операционная система или база данных SQL сервера. Внешняя система безопасности не сможет прочитать зашифрованную строку и доступ пользователя в систему не произойдет.

6. Выбрать Input Only, для того чтобы текст не отображался в текстовом поле, привязанном к объекту.

7. Нажать ОК

*Движки (Слайдеры)*

Можно создать объекты, которые пользователи могут перетаскивать/перемещать, и соответственно изменять значение привязанного к нему тега. Для этого используются анимационные связи раздела Slider. Объекту можно назначить анимационную связь

вертикального и горизонтального слайдера. Одному объекту можно назначить две анимационные связи горизонтального и вертикального слайдера одновременно. Таким образом, значения двух аналоговых тегов будет изменяться одновременно.

Для создания анимационной связи горизонтального слайдера

1. Нажать на объекте правой кнопкой мыши и в контекстном меню выбрать Animation Links. Появится соответствующее диалоговое окно.

2. В разделе Slider, нажать Horizontal. Появится окно Horizontal Slider.

3. В поле Tagname, ввести имя аналогового тега.

4. В разделе Properties, сделать следующее:

a. В разделе At Left End, ввести значение для тега, когда слайдер находится в крайнем левом положении.

b. В разделе At Right End, ввести значение для тега, когда слайдер находится в крайнем правом положении.

c. В разделе To Left, ввести количество пикселов, на которое слайдер может переместиться влево.

d. В разделе To Right, ввести количество пикселов, на которое слайдер может переместиться вправо.

5. Выберите Reference Location (положение курсора) на объекте, на котором будет фиксироваться курсор во время перемещения слайдера.

6. Нажать ОК.

Для создания анимационной связи вертикального слайдера

1. Нажать на объекте правой кнопкой мыши и в контекстном меню выбрать Animation Links. Появится соответствующее диалоговое окно.

2. В разделе Slider, нажать Vertical. Появится окно Vertical Slider.

3. В поле Tagname, ввести имя аналогового тега.

4. В разделе Properties, сделать следующее:

a. В разделе At Top, ввести значение для тега, когда слайдер находится в крайнем верхнем положении.

b. В разделе At Bottom, ввести значение для тега, когда слайдер находится в крайнем нижнем положении.

c. В разделе Up, ввести количество пикселов, на которое слайдер может переместиться вверх.

d. В разделе Down, ввести количество пикселов, на которое слайдер может переместиться вниз.

5. Выберите Reference Location (положение курсора) на объекте, на котором будет фиксироваться курсор во время перемещения слайдера.

6. Нажать ОК.

### Скрипты для Windows элементов управления

• Можно использовать встроенные функции в скриптах для того чтобы:

Получить или установить значение элемента управления.

• Разрешить, блокировать или спрятать элемент управления.

• Работать с элементами комбинированных списков, списков, текстовых полей и селективными кнопками.

Свойства элементов управления в режиме исполнения могут быть доступны только для чтения или и на запись и на чтение. Использовать функции GetPropertyX( ) и SetPropertyX() для управления и получения значений свойств.

Свойство .Value это стандартное свойство по умолчанию для всех мастеров Windows элементов управления в InTouch.Изменения, внесенные в данные свойство, синхронизируются в теге InTouch и мастерах Windows элементов управления.

*Поле .Value*

Стандартное свойство по умолчанию для всех Windows элементов управления.

*Категория*

Windows элементы управления

*Использование*

M, I, D – это версии функций GetProperty и SetProperty для Message, Integer, Discrete.

[ErrorNumber=]GetPropertyM("ControlName[.Value]", Tagname);

[ErrorNumber=]SetPropertyM("ControlName[.Value]",Value);

[ErrorNumber=]GetPropertyI("ControlName[.Value]",Tagname);

[ErrorNumber=]SetPropertyI("ControlName[.Value]",Value);

[ErrorNumber=]GetPropertyD("ControlName[.Value]",Tagname);

[ErrorNumber=]SetPropertyD("ControlName[.Value]",Value);

*Параметры*

ControlName

Имя элемента управления, например ChkBox\_4.

Tagname

Имя тега, содержащего целочисленное значение количества элементов списка.

[.Value] Опциональное свойство. Если не определено, то предполагается что используется свойство .Value.

Value Фактическое значение или тег InTouch (соответствующего типа), содержащий значение, которое будет записано после выполнения функция.

*Примечания*

Начальное значения тегов, привязанных к элементу управления список или комбинированный список, не могут быть использованы для инициализирования значений элементов управления список или комбинированный список. Данное свойство доступно на чтение и запись и во время разработки и в режиме исполнения. Если свойство .Value привязано к тегу либо в списке либо в комбинированном списке, то оно доступно только для чтения. Если свойство .Value привязано к тегу либо в группе селективной кнопки, полю текста или отмечаемой кнопке то оно доступно только для чтения и записи. Значение, которое определено в режиме исполнения, является значением по умолчанию в режиме исполнения.

*Тип данных*

Message (запись/чтение) для text box (поля текста), list boxes (списка) и combo box (комбинированного списка).

Integer (запись/чтение) для radio button (группы селективной кнопки).

Discrete (запись/чтение) для check box (отмечаемой кнопки).

*Примеры*

Следующий пример, устанавливает значение поля .Value группы селективной кнопки объекта RadioButton\_1 в 4: SetPropertyI( "RadioButton\_1.Value", 4 );

Сообщения об ошибках для Windows элементов управления

Функция wcErrorMessage() преобразует значение ошибки в текст, с описанием ошибки. Применяется к спискам, комбинированным спискам, полям текста, отмечаемым кнопкам и группам селективных кнопок. Функции, работающие с Windows элементами управления, возвращают результат работы. Данный результат используется для диагностики ошибок. Значения кода ошибок можно возвращать в тег типа Integer: ErrorNumber = wcGetItem("ControlName", Number, Tagname); В скрипте тег ErrorNumber будет содержать значение кода ошибки. Данное значение может быть передано в функцию wcErrorMessage(), которая возвратит текст описания ошибки. ErrorMsg = wcErrorMessge(ErrorNumber); В скрипте ErrorMsg это тег типа Message, содержащий текст возвращаемой ошибки. В таблице ниже приведено описание:

|  |  |
| --- | --- |
| **Ошибка** | **Описание** |
| -1 | Успешное завершение |
| -2 | Недостаточное количество памяти |
| -3 | Свойство доступно только для чтения |
| -4 | Указанный элемент уже существует |
| -5 | Имя объекта неизвестно |
| -6 | Имя свойства неизвестно |
| -Х | Неизвестная ошибка |

## Серверы ввода/выводы в InTouch

При функционировании InTouch - приложения в реальном времени информация обо всех его переменных хранится в базе данных. К такой информации относятся имя переменной, ее тип, минимальное и максимальное значения, уставки, способ отображения (дисплей, журнал) и т. д., а также информация о коммуникационных каналах, по которым происходит обмен данными между технологическим процессом и приложением.

InTouch - приложение поддерживает взаимодействие с DDE и OPC-серверами.

### Поддерживаемые коммуникационные протоколы

DDE (Dynamic Data Exchange - динамический обмен данными) представляет собой коммуникационный протокол, разработанный компанией Microsoft для обмена данными между различными Windows - приложениями. Этот протокол реализует взаимосвязи типа клиент - сервер между двумя одновременно исполняющимися программами.

В InTouch поддерживается также пакетированный DDE - обмен - FastDDE. Применение последнего заметно повышает эффективность и производительность обмена данными благодаря уменьшению общего количества DDE - пакетов, которыми клиент и сервер обмениваются между собой. Но принципиальные недостатки, связанные с надежностью и зависимостью от количества загруженных в текущий момент приложений Windows, остались. С целью расширения возможностей стандартного протокола DDE на локальную сеть компания Wonderware предложила NetDDE. Он позволяет приложениям, запущенным на объединенных в локальную сеть компьютерах, вести DDE - обмен. Позднее NetDDE лицензируется компанией Microsoft и поставляется в дистрибутивном пакете Windows. Следует отметить и то, что NetDDE допускает обмен информацией между приложениями на IBM PC и приложениями на машинах другого типа с операционной системой VMS или UNIX. Компания Wonderware предлагает и инструментальные средства для разработки DDE-серверов, в том числе и для не-Windows-платформ.

Протокол SuiteLink был специально разработан фирмой Wonderware для того, чтобы удовлетворить таким требованиям, как целостность данных, высокая производительность и простота диагностики. В основе протокола SuiteLink лежит протокол TCP/IP. SuiteLink не является заменой протоколам DDE, FastDDE и NetDDE. Новый протокол разработан для поддержания быстродействующих промышленных систем и обладает следующими характеристиками:

* Передача данных осуществляется в формате VTQ (Value, Time, Quality - значение, время, качество), в соответствии с которым каждая пересылаемая клиенту единица информации сопровождается метками времени и качества данных.
* Благодаря системному монитору операционной системы Windows NT (Performance Monitor) стал возможным расширенный анализ производительности по передаче данных, степени загрузки сервера, степени потребления ресурсов компьютера и сети, что особенно важно для проектирования и сопровождения больших распределенных промышленных сетей.
* Поддержка обмена данными между приложениями происходит независимо от того, исполняются ли эти приложения на одном узле сети или на разных.

Для реализации функций OPC - клиента Wonderware предлагает OPCLink - сервер, преобразующий OPC в SuitLink - протокол.

В материалах, предложенных компанией Wonderware, отмечается, что большинство реализованных OPC-серверов создают для каждого подключаемого к серверу клиента новый канал связи или нить. Для текущей обработки каждого клиента сервер должен переключаться между нитями. Каждая нить использует DCOM (Distributed Component Object Model) для организации обмена данными, и DCOM также управляет переключением нитей. В итоге возможна достаточно низкая производительность в сети.

Тесты, проведенные фирмой Wonderware, показали, что при обслуживании OPC-сервером 7 клиентов (при передаче 4 целых чисел в режиме обновления) сервер на 95% занимал ресурсы CPU. Это означает, что ресурсы компьютера практически целиком были заняты переключением нитей и DCOM- процедурами.

Поэтому на текущем этапе параметры производительности протокола SuiteLink превосходят параметры DCOM. Поставляемый в комплекте FactorySuite (Wonderware) OPCLink Server обеспечивает прием информации с OPC- сервера и передачу ее по протоколу SuiteLink в SCADA - систему InTouch и наоборот. Именно OPCLink Server рекомендуется устанавливать на одном узле с OPC- сервером, чтобы для сетевых передач использовался SuiteLink- протокол, а не DCOM (Рисунок 3.4.1 – 1).

|  |
| --- |
| http://www.bestreferat.ru/images/paper/39/13/5801339.gif *Рисунок 3.4.1 – 1 Использование SuiteLink - протокола в SCADA - системах.* |

Все описанные ниже особенности адресации распространяются и на OPC-серверы с одним лишь ограничением. При разработке InTouch - приложения создается канал связи с OPCLink - сервером (как с любым другим SuiteLink - сервером). Но рекомендуется использовать встроенный в InTouch OPC Browser для упрощения выбора параметров конфигурации подключаемого OPC - сервера.

### Особенности адресации

В InTouch вышеуказанные механизмы положены в основу обмена данными между приложениями InTouch и DDE и SuiteLink - серверами, которые, в свою очередь, связаны коммуникационными каналами с устройствами нижнего уровня (контроллерами).

Так как InTouch предназначен для разработки и поддержания интерфейса сбора данных и диспетчерского управления (Рисунок 3.4.2 – 1), среда исполнения WindowViewer при взаимодействии с контроллерным уровнем выступает, как правило, в роли приложения - клиента (узел View), запрашивающего данные у приложения - сервера (I/O Server).

|  |
| --- |
| http://www.bestreferat.ru/images/paper/40/13/5801340.gif *Рисунок 3.4.2 – 1 Обмен данными между InTouch - приложением*  *и технологическим процессом.* |

Через сервер ввода/вывода InTouch - приложение имеет возможность читать данные из контроллера или писать данные в него. Процесс обмена информацией InTouch - приложения с контроллером можно представить следующей схемой   
Здесь и встает один из главных вопросов организации обмена с серверами ввода/вывода: каким образом обеспечить клиенту доступ к запрашиваемой им информации?

Для организации обмена с приложением определяются каналы обмена или каналы доступа, характеризующиеся следующими параметрами:

* имя узла (Node Name);
* имя приложения ( Application Name );
* имя группы данных или топик (Topic Name );
* имя элемента ( Item Name ).

Имя приложения - это имя программы Windows, которая выполняет функции DDE, FastDDE, SuiteLink - серверов. Имя группы данных (топика) определяется при конфигурировании сервера на прием или передачу группы данных, которыми сервер будет обмениваться с контроллером или объединенными в сеть контроллерами. Определенные параметры группы (топика) зависят от конкретного сервера (поэтому рекомендуется изучать документацию и справочную систему выбранного сервера). Например, при использовании Modbus - сервера, позволяющего обеспечить взаимодействие с контроллером Modicon Micro 984 PLC, в качестве имени приложения (Application Name) должен быть Modbus, в качестве имени группы или топика (Topic Name) вводится любое имя (текстовая строка), но среди необходимых параметров группы из списка выбирается имя контроллера Modicon 984 PLC. А в качестве имени элемента (Item Name) следует выбирать название конкретного регистра контроллера (например, 40001 для контроллера Modicon Micro 984). Чтобы узнать правильный синтаксис имени элемента, необходимый для конкретных PLC, нужно обратиться к руководству по соответствующему серверу.

Определены все компоненты коммуникационного канала. С учетом введенных понятий схема обмена информацией для рассмотренного выше примера будет выглядеть следующим образом (Рисунок 3.4.2 – 2).

|  |
| --- |
| http://www.bestreferat.ru/images/paper/41/13/5801341.gif *Рисунок 3.4.2 – 2 Обмен информацией на примере Modbus - сервера.*  . |

Фирма Wonderware предлагает DDE и SuiteLink - серверы, которые поддерживают более 800 типов контроллеров основных производителей и различные протоколы.

Если нужного драйвера все-таки нет, можно воспользоваться пакетом разработки драйверов FactorySuite Toolkit.

Схемы, приведенные (Рисунок 3.4.2 – 2), интерпретируют стандартный обмен информацией между узлом (приложением) View и контроллером (ПЛК) в режиме сбора данных и управления. В этом режиме, как уже было сказано выше, приложение View - клиент по определению.

### Обмен данными с другими приложениями

Приложения InTouch могут взаимодействовать не только между собой, но и с другими Windows - приложениями. Одним из известных примеров такого приложения является Microsoft Excel. InTouch - приложение может считывать и записывать какие - либо значения в любую клетку открытой в Excel электронной таблицы. Аналогично и программа Excel может читать и записывать информацию в базу данных InTouch - приложения. Данный механизм обеспечивает одновременное обновление данных в одном приложении при изменении их значений в другом.

Если клиентом (приложением, запрашивающим информацию) по - прежнему является узел View, то Excel - это приложение, поставляющее информацию (сервер). В качестве группы или топика (Topic) тогда будет выступать имя таблицы Excel, а элемент обмена информацией - ячейка в таблице Excel (Таблица 3.4.3 - 1, вариант 1).

Когда клиентом является приложение Excel, а сервером - приложение View, группой в этом случае всегда является словарь переменных InTouch (база данных) с именем Tagname. Элементом обмена будет элемент базы данных - имя переменной (Таблица 3.4.3 - 1, вариант 2).

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *Таблица 3.4.3 - 1.*   |  |  |  |  | | --- | --- | --- | --- | | Приложение-клиент | Приложение-сервер | Группа | Элемент | | View | Exсel | Sheet1.XLS | R1C1 | | Excel | View | Tagname | R\_Level | |

В случае обмена данными по сети с использованием пакета Wonderware NetDDE необходимо к трехуровневой структуре адреса добавить четвертый уровень - имя удаленного узла сети (Node Name).

Подводя итог вышесказанному, следует подчеркнуть, что информация по доступу к данным устройств ввода/вывода или других приложений должна храниться в приложении (в словаре переменных). И разработчику в InTouch-приложении важно подключиться к вышеописанному каналу доступа. Для этого в InTouch необходимо определить имя доступа Access Name и связать его с переменной приложения.

### **Определение имени доступа в словаре переменных**

В InTouch - приложениях вся информация о переменных приложения хранится в Tagname Dictionary (Словарь переменных). Это не что иное, как база данных реального времени - один из центральных компонентов InTouch. При определении переменной в базе данных InTouch запрашивает определенную информацию о каждой переменной, например, имя переменной, ее тип, имя доступа и т. д.

В пакете InTouch используется два базовых типа переменных - Memory (внутренние) и I/O (переменные ввода/вывода).

Переменные типа Memory могут быть использованы для создания различных системных констант, моделирования элементов системы управления и в вычисляемых переменных, доступных другим Windows - программам.

Все переменные, которые получают или передают свое значение другой Windows - программе, должны иметь тип ввода/вывода (I/O). В эту категорию попадают переменные, которые посредством канала доступа (Access Name) принимают или отправляют данные из/в серверов ввода/вывода, других приложений InTouch, других программ Windows.

Определение новой переменной в базе данных InTouch, как и просмотр, и модификация атрибутов уже существующих переменных, производится в диалоге Tagname Dictionary (Рисунок 3.4.4 – 1). Доступ к этому диалогу осуществляется командой Speсial/Tagname Dictionary в окне среды разработки WindowMaker или двойным щелчком по иконке Tagname Dictionary в окне Application Explorer.

|  |
| --- |
| http://www.bestreferat.ru/images/paper/42/13/5801342.gif *Рисунок 3.4.4 – 1 Диалог Tagname Dictionary (Словарь переменных).* |

Поля Tagname и Comment предназначены для ввода имени переменной и соответствующего комментария. По умолчанию включена опция Read/Write (чтение/запись). Можно отметить и опцию Read Only, если в процессе исполнения WindowViewer должен только читать значение переменной.

В любое время в режиме проектирования можно открыть список переменных приложения щелчком по кнопке Select для выбора соответствующей переменной, просмотра списка или модификации атрибутов. Диалог Select Tag (выбор переменной) (Рисунок 3.4.4 – 2).

|  |
| --- |
| http://www.bestreferat.ru/images/paper/43/13/5801343.gif *Рисунок 3.4.4 – 2* *Диалог Select Tag (выбор переменной).* |

Для каждой переменной в этом диалоге приведена следующая информация: имя переменной, ее тип, имя доступа, группа аларма и комментарий.

Группа алармов (Alarm group, Рисунок 3.4.4 – 2) для переменной определяется в диалоге, вызываемом нажатием кнопки Group диалога Tagname Dictionary. Все, что касается алармов, рассматривается в соответствующем разделе ниже.

Выбор типа переменной осуществляется в диалоге Tag Types (тип переменной, Рисунок 3.4.4 – 3), вызываемом на экран нажатием кнопки Туре диалога Tagname Dictionary.

|  |
| --- |
| http://www.bestreferat.ru/images/paper/44/13/5801344.gif *Рисунок 3.4.4 – 3* *Диалог Tag Types (тип переменной)* |

В этом диалоге представлен полный список основных типов переменных InTouch. Выбор завершается отметкой соответствующей опции и щелчком по Ok. После выбора типа переменной программа возвращает пользователя в диалог Tagname Dictionary (Словарь переменных). При этом будет открыт и дополнительный диалог подробного описания переменной, содержание которого зависит от выбранного типа. Кнопка Access Name (имя доступа) используется для определения канала обмена (канала доступа) с сервером, с которым будет связана описываемая переменная. Имя доступа Access Name определяется именем узла, именем приложения и именем группы или топика. Имя топика должно совпадать с соответствующим именем, заданным при конфигурировании DDE, SuiteLink-сервера. Имя элемента, как компонента многоуровневого адреса, определяется в поле Item (Рисунок 3.4.4 – 4).

В распределенных системах InTouch имя доступа может быть определено либо как локальный адрес, либо как глобальный.

Локальные адреса используются в том случае, когда View - узлы имеют свои серверы ввода/вывода. На (Рисунок 3.4.4 – 4) узлы исполнения (View - узлы), каждый со своей копией одного и того же приложения, ссылаются на свои собственные источники данных ввода/вывода (серверы ввода/вывода).

|  |
| --- |
| http://www.bestreferat.ru/images/paper/45/13/5801345.gif  *Рисунок 3.4.4 – 4* *Сеть View – узлов с собственными серверами ввода/вывода* |

Поэтому при определении канала доступа к информации ввода/вывода достаточно трехуровневого адреса (Application - приложение, Topic - объект, Item - элемент). Имя узла (Node) в этом случае опускается. Щелчок по кнопке Access Name вызывает на экран одноименный диалог. Этот диалог предназначен для определения нового канала доступа (кнопка Add), модификации существующего (Modify) или удаления (Delete). Щелчок по кнопке Add вызывает диалог определения нового канала доступа. В качестве имени (канала) доступа (Access Names) рекомендуется выбирать имя группы или топика (Topic Name). Следует подчеркнуть, что поле Node Name (имя узла) оставлено пустым. Щелчок по кнопке Ok возвращает пользователя в диалог Access Names (имена доступа) с определенным именем доступа.

Глобальные адреса источников данных ввода/вывода позволяют нескольким View - узлам обращаться к одному и тому же серверу ввода/вывода. Такой подход предоставляет возможность отказаться от нескольких серверов ввода/вывода, однако менее защищен от отказов (Рисунок 3.4.4 - 5).

|  |
| --- |
| http://www.bestreferat.ru/images/paper/46/13/5801346.gif *Рисунок 3.4.4 – 5* *Архитектура с двумя View - узлами и сервером ввода/вывода* |

Два View - узла исполняют идентичные копии одного и того же приложения и ссылаются на один и тот же источник ввода/вывода (I/O сервер). Поэтому при определении канала доступа к информации ввода/вывода необходимо использовать четырехуровневый адрес (Node - узел, Application -приложение, Topic - объект, Item - элемент).

При выборе имени доступа действует то же правило, что и при локальной адресации: рекомендуется, чтобы это имя совпадало с именем группы данных или топика (Topic Name). Но поле Node Name (имя узла) необходимо заполнить. В качестве этого имени при глобальной адресации выбирают имя узла, на котором установлен сервер ввода/вывода, являющийся источником данных для нескольких приложений.

Для каждой переменной ввода/вывода задается атрибут Access Name. С одним именем доступа, как правило, связано большое количество переменных. Распределение переменных по группам (топикам) - произвольное. Но для оптимизации функционирования серверов рекомендуется в одну группу относить переменные с одинаковой частотой обновления. В противном случае частота, задаваемая при конфигурировании топика в сервере, должна соответствовать минимальному временному кванту. Желательно на этапе конфигурирования сервера определить группы (топики) для каждого частотного диапазона и в соответствии с этими группами создать имена доступа (Access Name) в InTouch (лучше даже, чтобы имена групп совпадали с именами доступа). А далее каждую описываемую в InTouch-приложении переменную типа I/O связывать с подходящим именем доступа для обеспечения рационального пакетирования данных.

## Тренды

Графическое представление значений технологических параметров во времени способствует лучшему пониманию динамики технологического процесса предприятия. Поэтому подсистема создания трендов и хранения информации о параметрах с целью ее дальнейшего анализа и использования для управления является неотъемлемой частью любой SCADA - системы.

Тренды реального времени (Real Time) отображают динамические изменения параметра в текущем времени. При появлении нового значения параметра в окне тренда происходит прокрутка графика справа налево. Таким образом текущее значение параметра выводится всегда в правой части окна.

Тренды становятся историческими (Historical) после того, как данные будут записаны на диск и можно будет использовать режим прокрутки предыдущих значений назад с целью посмотреть прошлые значения. Отображаемые данные тренда в таком режиме будут неподвижны и будут отображаться только за определенный период.

InTouch предлагает пользователю оба типа графических объектов, называемых трендами: тренд реального времени и исторический (архивный) тренд. Тренды реального времени дают возможность создавать графики изменения во времени четырех переменных (4 пера), в то время как для исторических трендов можно конфигурировать до восьми перьев в одном объекте. Количество объектов типа "тренд" в приложении, в том числе и в одном окне, не ограничено.

Оба типа трендов создаются c использованием специальных графических объектов инструментальной панели WindowMaker. InTouch также обеспечивает полный контроль над конфигурированием трендов. Для примера, можно определить диапазон времени, область значений, разрешение сетки, размещение временных отметок, число перьев и атрибуты цвета и т. д. Допускается переконфигурирование архивного тренда на этапе исполнения приложения (в Runtime).

### Архивирование значений переменной

При работе системы в режиме WindowViewer (среда исполнения) InTouch может производить запись значений переменных в регистрационный файл. Для того, чтобы архивирование переменной выполнялось, необходимо включить опцию Log Data (регистрация данных) при определении переменной в диалоге Tagname Dictionary.   
Запись в регистрационный файл производится всякий раз при изменении переменной на величину, превышающую порог для архивирования (Log Deadband), и по умолчанию один раз в час, если значение переменной за это время не изменилось. Поле Log Deadband находится в диалоге детального описания целой или вещественной переменной.

Чтобы значения переменных, для которых опция Log Data разрешена, записывались в регистрационные файлы, необходимо общее разрешение глобальной функции регистрации. Его задают в диалоге Historical Logging Properties (параметры архивирования, Рисунок 3.5.1 – 1), который вызывается на экран командой Special/Configure/Historical Logging. В этот диалог можно также войти из окна Application Explorer.

|  |
| --- |
| http://www.bestreferat.ru/images/paper/47/13/5801347.gif *Рисунок 3.5.1 – 1 Диалог Historical Logging Properties.* |

Включение опции Enable Historical Logging дает общее разрешение на регистрацию значений переменных. Срок хранения регистрационных файлов на диске (исключая текущий день) определяется в поле Keep Log Files for в днях. Если в это поле введено значение 0, файлы будут храниться бесконечно долго. Регистрационные файлы могут быть размещены в каталоге приложения (опция по умолчанию Store Log Files in Application Directory). В противном случае следует отметить опцию Store Log Files in Specific Directory (хранить файлы в ином каталоге) и ввести полный путь до каталога, в котором будут храниться регистрационные файлы (при работе с распределенными архивами - полный сетевой путь).

### Отображение

Тренды реального времени являются динамическими объектами. Они позволяют выводить изменения значений переменных, как только они происходят для любой конкретной переменной или для выражения, которое содержит одну или несколько переменных. Данные будут появляться в окне тренда и двигаться справа налево.

Чтобы создать тренд реального времени, необходимо:

* выбрать инструмент тренд реального времени в панели инструментов WindowMaker;
* щелкнуть в окне, затем переместить мышь по диагонали и сформировать прямоугольник необходимого размера;
* отпустить кнопку мыши, что вызовет появление тренда реального времени в окне (Рисунок 3.5.2 – 1).

|  |
| --- |
| http://www.bestreferat.ru/images/paper/48/13/5801348.gif *Рисунок 3.5.2 – 1 Объект «тренд реального времени».* |

При создании тренда реального времени настройки его конфигурации устанавливаются по умолчанию (настройки предыдущего тренда).

Для конфигурирования тренда реального времени следует либо дважды щелкнуть на созданном объекте, либо, предварительно выбрав объект, запустить команду Special/Animation Links. На экране появится диалог Real Time Trend Configuration (конфигурирование тренда реального времени). Среди настроек этого диалога можно отметить диапазон времени, охватываемый трендом (Time Span), частоту вывода значение переменной (Interval), разрешение сетки по большим и малым делениям горизонтальной и вертикальной осей (Time Division, Value Division), цвета фона и рамки графика (Color). Конфигурирование перьев тренда включает выбор имени переменной или выражения, цвета и толщины линии для каждого пера (поле Expression). Для повышения производительности системы следует отметить опцию Only update when in memory (обновлять, когда в памяти). В этом случае обновление данных тренда будет производиться только в моменты, когда окно с трендом отображается на дисплее (находится в RAM). Есть и другие способы повышения производительности при работе с трендами реального времени (уменьшение толщины линии графика, уменьшение частоты выводы значений переменной). Например, если установлен диапазон времени (Time Span) в 30 минут, а частота вывода - 2 секунды, то число измерений, которые нужно провести за каждые 30 минут, будет равно 900 (30 \* 60/2 = 900). При частоте выводе в 5 секунд число измерений существенно уменьшается: 30 \* 60/5 = 360. Исторические (архивные) тренды не являются динамическими. Они обеспечивают "снимок" состояния данных за прошедшее время, то есть по архивным данным. В отличие от трендов реального времени исторические тренды обновляются только по команде - при запуске скрипта, изменении значения выражения или нажатии оператором соответствующей кнопки. При конфигурировании архивного тренда можно создать "визиры" (ползунки, бегунки), с помощью которых удобно получить значения всех отображаемых переменных на один и тот же момент времени. Бегунки архивного тренда представляют собой позиционные индикаторы на временной оси, положение которых определяет объем извлекаемых данных. Связав объект "движковый регулятор" с полем бегунка, можно осуществлять перемещение вдоль архивного тренда. Кроме того, имеются функции вычисления среднего, минимального и максимального значений в определенном бегунком положении. Можно создать правый и левый бегунки и производить обработку данных кривой, расположенной между бегунками. Вычисляются следующие величины: среднее, минимальное, максимальное, отношение мин/макс и стандартное отклонение. В зависимости от положения бегунков на оси можно реализовать и другие функции (увеличение и уменьшение заключенной между бегунками области графика).

Благодаря системе распределенных архивов на один и тот же график можно выводить информацию из нескольких баз данных.

Все сказанное выше о механизме создания тренда реального времени инструментом Real Time Trend в среде разработки WindowMaker и о его последующем конфигурировании можно отнести и к архивному тренду, создаваемому инструментом Historical Trend среды разработки.

Предлагаемый ниже способ создания и конфигурирования архивного тренда предполагает использование мастер-средств библиотеки Wizard.

Нажатие кнопки выбора мастер-средств в панели инструментов вызывает появление на экране диалога Wizard Selection (выбор мастер-средств).   
После выбора из предложенного набора мастер-средств Hist Trend with Scooters (архивный тренд с бегунками) и щелчка по Ok программа возвращает пользователя в среду разработки. Курсор мыши при этом примет форму вставки . Последующий щелчок мыши на предполагаемом месте нахождения создаваемого объекта выводит на экран архивный тренд (Рисунок 3.5.2 – 2). Объекты этого типа ведут себя аналогично любым другим объектам, то есть их можно перемещать, масштабировать и т. д.

|  |
| --- |
| http://www.bestreferat.ru/images/paper/49/13/5801349.gif *Рисунок 3.5.2 – 2 Объект «архивный тренд».* |

Двойной щелчок на объекте приводит к появлению на экране диалога конфигурирования архивного тренда (Historical Trend Char Window).

|  |
| --- |
| http://www.bestreferat.ru/images/paper/50/13/5801350.gif *Рисунок 3.5.2 – 3 Диалог конфигурирования архивного тренда.* |

Для конфигурирования тренда с параметрами по умолчанию следует нажать кнопку Suggest (вариант). Нажатие кнопок Times и Values выводит на экран окна конфигурирования разрешения сетки по большим и малым делениям горизонтальной и вертикальной осей, цвета фона и рамки графика, временного диапазона и т. д. Кнопка Pens (перья) предназначена для настройки перьев архивного тренда.

Чтобы добавить в тренд функции масштабирования и перемещения или элементы управления перьями, следует использовать панели Zoom/Pan и Trend Pen Legend (Рисунок 3.5.2 – 3), соответственно. Для того, чтобы эти компоненты работали совместно, они должны иметь одинаковые имена (Hist Trend).

### Изменение параметров в режиме исполнения

При управлении в режиме реального времени оператор анализирует архивную информацию. Объем информации, ее временные диапазоны, объем статистических данных, необходимые для принятия решения по управлению технологическим процессом, заранее не известны. Поэтому оператор должен иметь возможность менять настройки архивных трендов, не выходя из режима Runtime. В InTouch такая возможность существует.

Для этого следует включить опцию Allow runtime changes (разрешить изменения во время исполнения) в диалоге конфигурирования архивного тренда (в книге не показан).

Теперь в режиме WindowViewer щелчок на архивном тренде будет вызывать на экран диалог изменения параметров архивного тренда (Historical Trend Setup). В этом диалоге можно определить дату и время начала архивного тренда (поле Chart Start), его временной диапазон (Chart Length), присвоить перьям цвет и имена переменных, выбирая их из словаря.

Архивный тренд может выводиться в одном из трех возможных режимах:

* Min/Max - график изменения значений переменной в виде вертикальных линий в процентах от всего диапазона, позволяющий оценить скорость изменения переменной;
* Average/Scatter - график среднего значения переменной;
* Average/Bar Chart - график среднего значения переменной в виде гистограммы.

Выбор режима производится в поле Display Mode.

### Система распределенных архивов

В InTouch имеется система распределенных архивов, обеспечивающая поиск архивных данных в любом InTouch - приложении. Данная система расширяет возможности стандартных архивов InTouch, позволяя одновременно получать информацию из нескольких удаленных баз данных, которые в этом случае называются провайдерами архивов.

Одновременно можно обращаться к восьми провайдерам (по одному на каждое перо). Каждый узел, выполняющий функцию регистрации, может писать только в один архив.

Система (Рисунок 3.5.4 – 1), имеет два провайдера архивов. Левый провайдер регистрирует информацию только из узла, расположенного слева внизу. Правый провайдер регистрирует информацию из узла, расположенного справа вверху. Остальные три узла (вверху слева) лишь используют архивные данные. Читать информацию из архивных файлов может каждый из узлов системы.

Создание такой системы предполагает следующие действия:

* создание списка провайдеров архивов;
* создание и определение параметров объекта "архивный тренд";
* конфигурирование приложения на удаленное архивирование данных;
* копирование приложения на все узлы.

|  |
| --- |
| http://www.bestreferat.ru/images/paper/51/13/5801351.gif *Рисунок 3.5.4 – 1 Распределенная система архивов.* |

## Встроенные языки программирования

Встроенные языки программирования - мощное средство SCADA - систем, предоставляющее разработчику гибкий инструмент для разработки сложных приложений. Первые версии SCADA - систем либо не имели подобных языков, либо эти языки реализовывали небогатый набор функций. В современных версиях SCADA - систем функциональные возможности языков становятся существенно богаче. Явно выделяются два подхода:

* Ориентация встроенных языков программирования на технологов. Функции в таких языках являются высокоуровневыми, не требующими профессиональных навыков программирования при их использовании. Количество таких функций в базовых поставках не исчисляется сотнями, хотя существуют свободно распространяемые библиотеки дополнительных функций.
* Ориентация на системного интегратора. В этом случае в качестве языков чаще всего используются VBasic - подобные языки.

В каждом языке допускается расширение набора функций. В языках, ориентированных на технологов, это расширение достигается с помощью дополнительных инструментальных средств (Toolkits). Разработка дополнительных функций выполняется обычно программистами - профессионалами.

Разработка новых функций при втором подходе выполняется обычно разработчиками приложений (как и в традиционных языках программирования).

Полнота использования возможностей встроенных языков (особенно при втором подходе) требует соответствующего уровня квалификации разработчика, если, конечно, в этом есть необходимость. Требования задачи могут быть не столь высокими, чтобы применять всю "мощь" встроенного языка.

Во всех языках функции разделяются на группы, часть из которых присутствует практически во всех языках: математические функции, функции работы со строками, обмен по SQL , DDE - обмен и т. д.

В разрабатываемом приложении создаются программные фрагменты, состоящие из операторов и функций языка, которые выполняют некоторую последовательность действий. Эти программные фрагменты связываются с разнообразными событиями в приложении, такими как нажатие кнопки, открытие окна, выполнение логического условия (a +b > c). Каждое из событий ассоциируется с графическим объектом, окном, таймером, открытием/ закрытием приложения. Когда приложение содержит сотни окон, тысячи различных графических объектов, а с каждым из них связано несколько событий, в приложении может "работать" огромное количество отдельных программных фрагментов. Велика вероятность их "одновременной" активизации.

Каждая из функций во встроенном языке выполняется в синхронном или асинхронном режиме. В синхронном режиме выполнение следующей функции не начинается до тех пор, пока не завершилось исполнение предыдущей. При запуске асинхронной функции управление переходит следующей, не дожидаясь завершения исполнения предыдущей функции.

В связи с этим возникает несколько вопросов. С каким приоритетом исполняется каждый из фрагментов, допускается ли рекурсия при обработке событий и если да, то каков уровень вложенности? В SCADA - системах уровень вложенности пока не стандартизован, но оговаривается особо в рамках каждой из них.

### Скрипты

Скрипты в InTouch - это программные фрагменты, активизируемые по событиям (по нажатию клавиши, кнопки, открытию окна, изменению значения переменной и т. д.).

В InTouch различают несколько типов скриптов:

* Application Scripts (скрипты уровня приложения) относятся ко всему приложению и используются для запуска других приложений, имитации технологических процессов, вычисления значений переменных и т.д.
* Window Scripts (скрипты уровня окна) связываются с конкретным окном.
* Key Scripts (клавишные скрипты) привязываются к какой-либо клавише или комбинации клавиш клавиатуры. Это может быть полезным при создании каких-либо глобальных для всего приложения функций (возврат в главное окно, окончание сеанса работы с приложением и т. д.).
* Touch Pushbutton Action Scripts (скрипты, запускаемые кнопками) очень похожи на клавишные скрипты и связываются с объектами, которые будут использоваться в качестве исполнительных кнопок. Эти скрипты запускаются при каждом нажатии на объект-кнопку.
* Condition Scripts (скрипты по изменению логического выражения) связываются с логической переменной или выражением, которое будет принимать значения либо "истина", либо "ложь". Логические скрипты могут содержать в себе и аналоговые переменные.
* Data Change Scripts (скрипты по изменению данных) связываются либо с переменной, либо с полем переменной. Эти скрипты исполняются только один раз, когда значение переменной либо поля меняется на величину, превышающую значение допуска, заданного в словаре переменных.
* ActiveX Event (скрипты событий ActiveX) предназначены для поддержки механизма реакции на события в ActiveX - объектах. С каждым событием может быть связан один скрипт типа ActiveX Event, запускающийся в WindowViewer во время исполнения приложения.
* Quick Function - скрипты, которые могут вызываться из других скриптов и использоваться в выражениях при определении динамических свойств объектов.

Диалоги редактора, открываемые при создании скриптов различных типов, имеют небольшие отличия. Вызов диалога редактора скриптов в окне WindowMaker осуществляется командой Special/Scripts с последующим выбором типа создаваемого или редактируемого скрипта. Для этого можно также воспользоваться окном Application Explorer, выбрав папку Scripts.

Редактор скриптов InTouch поддерживает два типа скриптов: простые и сложные. Простые скрипты - это скрипты, содержащие операторы присваивания, сравнения, простые математические функции и т. д. Сложные скрипты позволяют выполнять различные логические операции типа IF - THEN - ELSE, а также могут включать циклы типа FOR - NEXT.

Справа, в поле Functions, размещены клавиши вызова списков различных групп встроенных функций. Доступ к спискам встроенных функций возможен также командой Insert/Functions с последующим выбором группы функций.

### Встроенные функции

В пакете InTouch имеется набор встроенных функций, которые могут быть связаны с командами или использованы в скриптах для выполнения самых различных задач.

Все встроенные функции разбиты на четыре группы:

* String...- для обработки различных символьных строк и переменных;
* Math...- математические функции;
* System...- системные функции;
* Misc...- функции для работы с алармами распределенных систем, трендами, печатью и др.

Вызов списка функций группы осуществляется нажатием соответствующей клавиши.

### Функции Quick Functions

Quick Functions - это скрипты, которые могут вызываться из других скриптов и использоваться в выражениях при определении динамических свойств объектов. Скрипты Quick Functions хранятся внутри того приложения, в котором они были созданы, и могут многократно использоваться в других скриптах InTouch.   
Наиболее часто эти функции используют в выражениях при определении динамических свойств объектов. Чем это вызвано? Дело в том, что длина выражения в поле Expression диалогов определения динамических свойств объектов должна быть не более 256 символов. Это относится к таким динамическим свойствам, как цвет линии, цвет заполнения, изменение высоты и ширины, вертикальное и горизонтальное перемещение, вертикальное и горизонтальное заполнение, видимость, мерцание, ориентация, блокировка.

Для ввода более длинных выражений можно воспользоваться функциями Quick Functions. При этом выражение в поле Expression должно содержать операторы CALL вызова функций Quick Functions, каждая из которых, в свою очередь, должна иметь в качестве последнего оператора RETURN для возврата результата в вызывающее выражение. Организованное таким образом выражение может содержать многие тысячи символов и быть сколь угодно сложным.

Сохраненная функция Quick Functions может быть использована в любом другом скрипте или выражении.

Quick Functions могут быть синхронными и асинхронными скриптами. Синхронные скрипты выполняются последовательно, в то время, как после запуска одного асинхронного скрипта может быть запущен другой (синхронный или асинхронный) скрипт. Это позволяет отделять исполняющиеся довольно долго операции (типа обращений к базам данных) от основной программы. Асинхронные скрипты не могут возвращать результаты. Поэтому в качестве скриптов Quick Functions, используемых в выражениях (Expression) для определения динамических свойств объектов, следует применять только синхронные скрипты.  
Создание скриптов Quick Functions осуществляется в диалоговом окне редактора Quick Functions. Вызов этого диалога на экран в окне WindowMaker производится в командой Special/Scripts с последующим нажатием на строке Quick Functions.

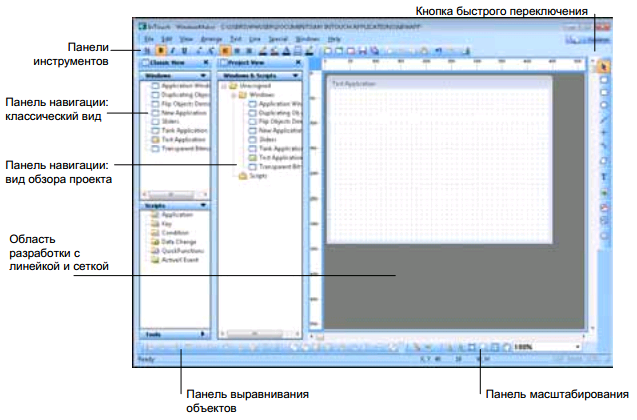
Список Name содержит имена всех определенных к данному моменту скриптов Quick Functions. Щелчок по имени скрипта выводит его текст в рабочее поле диалога.

Команда Scripts/New предназначена для создания нового скрипта и вызывает на экран диалог для ввода его имени. После щелчка по Ok новое имя будет включено в список имен Name.

Следующий этап - определение аргументов нового скрипта в таблице Arguments диалога Quick Function. В левую колонку таблицы вводят имя аргумента (до 31 символа), в правую - его тип (Integer, Real, Discrete, Message). В одном скрипте допускается до 16 аргументов.

После определения типов аргументов можно приступать к написанию текста скрипта Quick Function в рабочем поле (под таблицей Arguments).

# ПРИЛОЖЕНИЕ А

*Рисунок А.1 – Вид окна WindowMaker*

# Лист регистрации изменений

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Изм. | Номера листов | | | | Всего листов в документе | № документа | Подпись | Дата |
| Измененных | Замененных | Новых | Аннулированных |
| 1 |  |  |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |  |  |
| 4 |  |  |  |  |  |  |  |  |
| 5 |  |  |  |  |  |  |  |  |
| 6 |  |  |  |  |  |  |  |  |
| 7 |  |  |  |  |  |  |  |  |
| 8 |  |  |  |  |  |  |  |  |
| 9 |  |  |  |  |  |  |  |  |
| 10 |  |  |  |  |  |  |  |  |
| 11 |  |  |  |  |  |  |  |  |
| 12 |  |  |  |  |  |  |  |  |
| 13 |  |  |  |  |  |  |  |  |
| 14 |  |  |  |  |  |  |  |  |
| 15 |  |  |  |  |  |  |  |  |
| 16 |  |  |  |  |  |  |  |  |
| 17 |  |  |  |  |  |  |  |  |
| 18 |  |  |  |  |  |  |  |  |
| 19 |  |  |  |  |  |  |  |  |
| 20 |  |  |  |  |  |  |  |  |
| 21 |  |  |  |  |  |  |  |  |
| 22 |  |  |  |  |  |  |  |  |
| 23 |  |  |  |  |  |  |  |  |
| 24 |  |  |  |  |  |  |  |  |
| 25 |  |  |  |  |  |  |  |  |
| 26 |  |  |  |  |  |  |  |  |
| 27 |  |  |  |  |  |  |  |  |
| 28 |  |  |  |  |  |  |  |  |
| 29 |  |  |  |  |  |  |  |  |
| 30 |  |  |  |  |  |  |  |  |
| 31 |  |  |  |  |  |  |  |  |
| 32 |  |  |  |  |  |  |  |  |
| 33 |  |  |  |  |  |  |  |  |
| 34 |  |  |  |  |  |  |  |  |