Возможности подключения к серверу (для программиста 1С WEB компонента)

Внимание! Подключаемая компонента поддерживает только 32-битные версии платформы 1С.

Регистрация и подключение библиотеки

Регистрация библиотеки AddIn_1CToWebServer.dll - reg.bat:

```
@ECHO OFF
REM The following directory is for .NET 4.0
set DOTNETFX4=%SystemRoot%\Microsoft.NET\Framework\v4.0.30319
set PATH=%PATH%;%DOTNETFX4%
echo ----------------------------------------------------
regasm.exe  "AddIn_1CToWebServer.dll" /tlb /codebase
echo ----------------------------------------------------
```

Регистрации библиотеки производится от имени Администратора. В примере “Тест ВК WEB сервера” регистрация производится автоматически при первом запуске программы.

Пример подключения компоненты AddIn_1CToWebServer.dll:

```
Если глВесы1 = Неопределено Тогда
        Если не ПодключитьВнешнююКомпоненту("AddIn.1CToWebServer") Тогда
            Сообщить("Компоненту AddIn.1CToWebServerне удалось подключить,
требуется инициализация!!!");
            Возврат Ложь;
        Иначе
            Попытка
      глВесы1 = Новый("AddIn.1CToWebServer");
            //Сообщить("Компонента AddIn.1CToWebServer загружена");
                    Возврат Истина;
            Исключение
      глВесы1 = Неопределено;
            Сообщить("Не удалось загрузить компоненту");
            Сообщить(""+ОписаниеОшибки());
            Возврат Ложь;
```

```
                КонецПопытки;
            КонецЕсли;
КонецЕсли;
```

Основные функции:

ОткрытьСерверВесы(<АдресСервера>, <debugMode>)
Подключиться к WEB Серверу АВТО.
<АдресСервера> – сетевой путь (IP адрес) к компьютеру, где установлен WEB сервер (например 192.168.10.122:8090) <debugMode> - логическое значение режим отладки. После запуска включается цикл опроса сервера **(Включен = 1)** . Если WEB-Сервер по указанному адресу откликнулся, то проверить состояние можно с помощью свойства **СерверОткрыт = 1**

ЗакрытьСерверВесы()
Закрыть соединение с сервером.
После запуска выключается цикл опроса сервера **(Включен = 0)** . Проверить состояние можно с помощью свойства **СерверОткрыт = 0**

Включен
Свойство - флаг состояния активности цикла опроса WEB-сервера **Включен=1** – цикл включен

СерверОткрыт
Свойство - флаг состояния подключения к WEB-серверу **СерверОткрыт=1** – сервер открыт

Внешние события компоненты:

Источник = «1CToWebServer»

Наименования событий
Событие = «error» - ошибка
Событие = «ServerOpen» - сервер открыт
Данные – Адрес Сервера
Событие = «ServerClose» – сервер закрыт
Данные – Адрес Сервера Событие = «Start» – старт цикла опроса
Событие = «Stop» – остановка цикла опроса WEB сервера
Событие = «ServerUpdateParam» – принят обновлен очередной пакет с параметрами
Событие = «ServerUpdatePhoto»– приняты обновлены фотоснимки с камер

События оборудования ВЕСЫ
Событие = UpdateMassa – изменено значение массы
Событие = UpdateFreq – изменено значение частоты передачи с весового индикатора
Событие = UpdatefSTABIL – изменено состояние стабильности
Событие = UpdateStateVesyName – изменено состояние весов
Событие = UpdateAutoVsv AutoVsv_UpdateDateTime – изменено значение даты последнего автовзвешивания
Событие = UpdateAutoVsvSelect – автовзвешивание загружено

События системы распознавания

Событие = UpdateCamRecognTS – изменилось состояние номер найден (для всех камер)
Событие = UpdateCamNumbTS – изменилось значение найденного номера транспорта (для всех камер)
Событие = UpdateCamNumbPricep – изменилось значение найденного номера прицепа (для всех камер)
Событие = UpdateCam1RecognTS – изменилось значение найденного номера транспорта в камере 1
Событие = UpdateCam2RecognTS – изменилось значение найденного номера транспорта в камере 2
Событие = UpdateCam3RecognTS – изменилось значение найденного номера транспорта в камере 3
Событие = UpdateCam4RecognTS – изменилось значение найденного номера транспорта в камере 4

Свойства и команды оборудования - ВЕСЫ:

Масса
Функция возвращает значение массы.

Частота
Функция возвращает частоту пакетов, посылаемых весовым индикатором в компьютер

УстановитьНоль()
Послать команду установки нуля на весовой индикатор.

Ноль
Функция возвращает флаг “Ноль” (значение 1 или 0)

Стабил *вызывается перед взвешиванием, для проверки*
Функция возвращает флаг “Стабильность” (значение 1 или 0)

ИмяВесов
Функция возвращает имя весов

ЭмуляцияПоказаний
Свойство, возвращает 1 если включен режим эмуляции

ВходПользователя (<ИмяПользователя>)
Регистрирует вход пользователя в базу событий Сервера Весы <ИмяПользователя> – Строка. Имя пользователя, подключившегося к весам

ВыходПользователя (<ИмяПользователя>)
Регистрирует выход пользователя в базу событий Сервера Весы <ИмяПользователя> – Строка. Имя пользователя, подключившегося к весам

ИмяСостояниеВесов
Возвращает строку сокращенного наименования состояния весов

РасшифровкаСостояниеВесов
Возвращает строку расшифровки состояния весов

Функции контроля взвешивания:

ЗафиксироватьВзвешивание(<GUIDДокумента>, <МАССА>, <ИмяПользователя>);

(ОБЯЗАТЕЛЬНО ДЛЯ ИСПОЛЬЗОВАНИЯ)

Устанавливает флаг контроля, и сообщает Серверу о том, что взвешивание произведено. Следующая за ней команда **ЗаписатьДокументВзвешивания**.
GUIDДокумента – уникальный идентификатор взвешивания. Если пуст, то заполняется новым значением. **МАССА** – масса взвешенного груза
ИмяПользователя – Строка. Имя пользователя.

ПослеВзвешиванияНеСъезжал() *вызывается перед взвешиванием, для проверки съехал ли транспорт с весов.*
Функция возвращает флаг «ПослеВзвешиванияНесъезжал» (значение 1 или 0)

ВзвешиваниеНеСНоля() *вызывается перед взвешиванием, для проверки было ли обнуление весов перед заездом транспорта.*
Функция возвращает флаг «ВзвешиваниеНеСНоля» (значение 1 или 0)

Функции выгрузки документа взвешивания в дублирующий журнал Сервера Весы:

ЗаписатьДокументВзвешивания(<GUIDДокумента>, <КодТипаВзвешивания>, <ТипВзвешивания>,<НомерДокумента>,<ДатаДокумента>,<ВремяДокумента>,<КодОтправитель>,<Отправитель>,<КодПолучатель>,<Получатель>,<КодПунктОтправления>,<ПунктОтправления>,<КодПунктНазначения>,<ПунктНазначения>,<КодГруз>,<Груз>,<КодВодитель>, <Водитель>,<НомерНакладной>,<ДатаНакладной>,<ТараПоДокументу>,<БруттоПоДокументу>,<НеттоПоДокументу>,<КодМаркаТС>,<МаркаТС>,<НомерТС,РегионТС>, <НомерПрицепа>,<РегионПрицепа>,<ИмяПользователя>, ТАРА>, ДатаТарирования>, <ВремяТарирования>,<БРУТТО>,<ДатаБрутирования>, <ВремяБрутирования>, <НЕТТО>, <флагУдален>)

Создает или обновляет в базе взвешиваний запись о взвешивании с заполненными полями. Если GUID (*уникальный идентификатор взвешивания*) не найден создает новую запись, иначе обновляет существующую
GUIDДокумента – уникальный идентификатор взвешивания, тип строка GUID. Если <GUIDДокумента> пустой, создается при первой записи сервером Firebird.
ТипВзвешивания – Строки - наименование типа взвешивания.
НомерДокумента – Номер документа.
ДатаДокумента – дата документа.
ВремяДокумента – время документа.
КодОтправителя, Отправитель – Код и наименование Контрагенты.
КодПолучатель, Получатель – Код и наименование Контрагенты.
КодПунктОтправления, ПунктОтправления – Код и наименование Пункты или

МестаХранения.
Код ПунктНазначения, ПунктНазначения – Код и наименование Пункты или МестаХранения.
Код Груз, Груз – Код и наименование Номенклатура.
КодВодитель, Водитель – Код и ФИО водителя.
НомерНакладной – строка НомерНакладной.
ДатаНакладной – Дата накладной.
ТараПоДокументу – тара по документу.
БруттоПоДокументу – брутто по документу.
НеттоПоДокументу – нетто по документу.
Код МаркаТС, МаркаТС – Код и наименование Марки ТС.
НомерТС,РегионТС – Гос.номер и регион ТС.
НомерПрицепа – Гос. Номер прицепа.
РегионПрицепа – регион прицепа.
ИмяПользователя – Строка. Имя пользователя.
ТипТары – Строка. Характер значения тары – “взвешивание” или “вручную”.
ТАРА, ДатаТарирования,ВремяТарирования – Строка масса тары, дата и время тарирования.
ТипБрутто – Строка. Характер значения брутто – “взвешивание” или “вручную”.
БРУТТО, ДатаБрутирования,ВремяБрутирования – Строка масса брутто, дата и время брутирования.
НЕТТО – Масса НЕТТО.
флагУдален – признак флага удален (‘1’ – удален, ‘0’ – не удален).

поля <Код..> используются в базе взвешиваний для ведения справочников.

ЗаписатьПараметрДокументаВзвешивания(<GUIDДокумента>, <ИмяПоля>,<Значение>)
Создает или обновляет в базе взвешиваний запись о взвешивании
Обновляет значение указанного поля.
Если GUID (уникальный идентификатор взвешивания) не найден создает новую запись, иначе обновляет существующую
GUIDДокумента – уникальный идентификатор взвешивания, тип строка GUID. Если <GUIDДокумента> пустой, создается при первой записи сервером Firebird.

ЗаписатьСобытие(<ИсточникСобытия>,<Описание>)
Создает в фискальном журнале событий записать о дополнительном событии. Регистрация с текущим временем.
Возвращает 1-если успешно, 0-неудачно
<ИсточникСобытия> – метка (целое число) указывающая на источник события
2 - Оператор
3 - Взвешивание
4 - Программа
5 - Администрирование
<Описание> – Строка. Описание события

Функции работы с видеокамерами:

JPG полном размере
ПолучитьФотоКамеры1();

JPG в размере окна
ПолучитьФотоПоРазмеруКамеры1(<Ширина>,<Высота>);

БМП в полном размере
ПолучитьФотоБМПКамеры1();

БМП в размере окна
ПолучитьФотоБМППоРазмеруКамеры1(<Ширина>,<Высота>);

Пример Сохранения фото в файл:

```
БинДанные =   Base64Значение(ТекущиеВесы.ПолучитьФотоКамеры1())
Фото1 = Новый Картинка(БинДанные);
Фото1.Записать(КаталогИБ+ "\PHOTO\"+GUIDДокументаВзвешивания +
"_PHOTO_"+НомерВзв+"1.jpg");
```

Функции системы распознавания номеров:

флаг номер найден
КамераНайденНомер свойство возвращает – 1, если номер найден

Получить НомерТС
КамераНомерТС возвращает строку с номером

Получить РегионТС
КамераРегионТС возвращает строку с регионом

Получить НомерПрицепа
КамераНомерПрицеп возвращает строку с номером

Получить РегионПрицепа
КамераРегионПрицеп возвращает строку с регионом

Функции оборудования УДВВ и контроля проезда транспорта:

РасшифровкаСостоянияПроезда
Строка расшифровка текущего состояния проезда

СобытиеНомерВКамере1
Возвращает 0 или 1. Событие – стабильное распознавание номера в камере 1

СобытиеНомерВКамере2
Возвращает 0 или 1. Событие – стабильное распознавание номера в камере 2

СобытиеНомерВКамере3
Возвращает 0 или 1. Событие – стабильное распознавание номера в камере 3

СобытиеНомерВКамере4
Возвращает 0 или 1. Событие – стабильное распознавание номера в камере 4

СобытиеДатчик1
Возвращает 0 или 1. Событие – сработал датчик 1

СобытиеДатчик2
Возвращает 0 или 1. Событие – сработал датчик 2

РазрешитьПроезд
Возвращает 0 или 1. Флаг – проезд разрешен

РазрешитьПроезд1
Возвращает 0 или 1. Флаг – проезд разрешен в направлении 1

РазрешитьПроезд2
Возвращает 0 или 1. Флаг – проезд разрешен в направлении 2

УДВВ_Частота
Возвращает частоту обмена с устройством ввода-вывода

УДВВ_СостояниеВыходы
Состояние выходов устройства ввода-вывода

УДВВ_СостояниеВходы
Состояние входов устройства ввода-вывода

УДВВ_УстановитьВыход(int NBit, int _Value)
Команда установки значения выхода, нужного бита

УстановитьРазрешитьПроезд()
Команда установить флаг проезд разрешен

УстановитьРазрешитьПроезд1()
Команда установить флаг проезд разрешен в направлении 1

УстановитьРазрешитьПроезд2()
Команда установить флаг проезд разрешен в направлении 2

Функции загрузки автоматических взвешиваний, создаваемых Сервером ВЕСЫ Авто:

Сервер Весы Авто при переходе в автоматический режим создает записи взвешиваний независимо от 1С. Данный набор функций позволяют последовательно скопировать данные автовзвешивания начиная с первой с помощью последовательности команд:

1. Анализ количество новых автовзвешиваний:

```
Если ГлВесы.КоличествоНовыхАвтоВзвешиваний > 0 Тогда
```

2. Выбрать первое

```
ГлВесы.ВыбратьАвтоВзвешивание()
```

3. Дождаться когда будет выбрано

```
Если ГлВесы.АвтовзвешиваниеВыбрано = 1 Тогда
```

4. Читать параметры выбранного автовзвешивания в новый документ:

```
GUIDДокументаВзвешивания = ГлВесы.ПолучитьПараметрАвтоВзвешивания("CODE");
        ДатаДок = ГлВесы.ПолучитьПараметрАвтоВзвешивания("DATE");
ВремяДок = ГлВесы.ПолучитьПараметрАвтоВзвешивания("TIME");
                    ***** и т.д.
```

5. Записать новое взвешивание:

```
ГлВесы.ЗаписатьДокументВзвешивания(…
```

После этого сбрасывается **ГлВесы.АвтовзвешиваниеВыбрано = 0**, далее по новой.

флаг состояние режима автовзвешивания
РежимАвтоВзвешивания – возвращает 0 или 1

команда установить режим автовзвешивания
УстановитьРежимАвтоВзвешивания(Режим) <Режим> – 0 или 1

количество незагруженных автовзвешиваний
КоличествоНовыхАвтоВзвешиваний – возвращает целое число от 0 и больше.

Команда выброра первого автовзвешивания
ВыбратьАвтоВзвешивание() – результат не возвращает

флаг готовности для загрузки выбранного автовзвешивания
АвтовзвешиваниеВыбрано – возвращает 0 или 1

получить параметр загруженного автовзвешивания
ПолучитьПараметрАвтоВзвешивания(ИмяПараметра) <ИмяПараметра> – 0 или 1

Получить фото первого отвеса загруженного автовзвешивания
ПолучитьФОТОАвтоВзвешОтвес1(НомерФото) <НомерФото> – номер камеры 1, 2, 3 или 4

Получить фото второго отвеса загруженного автовзвешивания
ПолучитьФОТОАвтоВзвешОтвес2(НомерФото) <НомерФото> – номер камеры 1, 2, 3 или 4

```
//-----------------------------------------------------------------------------
---
```

```
// Алгоритм загрузки новых взвешиваний из Сервера Весы
//-----------------------------------------------------------------------------
---
Процедура ОбработкаОжиданияВК()

//-----------------------------------------------------------------
          //----- Загрузка автовзвешиваний, фоновый процесс -----------------
 //-----------------------------------------------------------------
          Если (ДатаВремяАвтоВзвешивания <> Компонента.ДатаВремяАвтоВзвешивания)
Тогда
       ДатаВремяАвтоВзвешивания = Компонента.ДатаВремяАвтоВзвешивания;
       АвтоВзвешиваниеВыбрано_ = Компонента.АвтоВзвешиваниеВыбрано;

               Если (Компонента.КоличествоНовыхАвтоВзвешиваний > 0) Тогда
                   Если (АвтоВзвешиваниеВыбрано_ = 0)
                       Тогда   Компонента.ВыбратьАвтоВзвешивание()
                       Иначе СоздатьАвтоВзвешивание();
                  КонецЕсли;
               КонецЕсли;
          КонецЕсли;
          Если АвтоВзвешиваниеВыбрано_ <> Компонента.АвтоВзвешиваниеВыбрано Тогда
       АвтоВзвешиваниеВыбрано_ = Компонента.АвтоВзвешиваниеВыбрано;
               Если (АвтоВзвешиваниеВыбрано_ = 1) и
(Компонента.КоличествоНовыхАвтоВзвешиваний>0) ТОгда
          СоздатьАвтоВзвешивание();
               КонецЕсли;
          КонецЕсли;

ПодключитьОбработчикОжидания("ОбработкаОжиданияВК", 0.2, Истина);
КонецПроцедуры
```

From:
http://docuwiki.vesysoft.ru/ - **База знаний**

Permanent link:
http://docuwiki.vesysoft.ru/doku.php?id=manager:1cweb

Last update: **2017/12/18 11:40**