

## Автоматизированное тестирование в 1С:Предприятии 8.3

В платформе “1С:Предприятие” 8.3 появился новый механизм, предназначенный для имитации интерактивных действий пользователей системы – автоматизированное тестирование. Автоматизированное тестирование не поддерживает работу с обычным интерфейсом, а только с управляемым.

При тестировании используются два вида клиентских приложений – менеджер тестирования и клиент тестирования. Менеджер тестирования устанавливает связь с клиентом тестирования и выполняет сценарий тестирования.

Сценарий тестирования – это код на встроенном языке, в котором описывается последовательность выполняемых интерактивных действий.

Для этого во встроенный язык добавлены новые объекты, которые на абстрактном уровне описывают интерфейс приложения (оперируя понятиями окна, формы, элементов управления и т.п.), а также описывают действия пользователей (навигация по конфигурации, ввод данных и т.п.).

Менеджер тестирования может быть толстым или тонким клиентом. Клиент тестирования – толстым, тонким клиентом или веб-клиентом. Менеджер тестирования может быть подключен к нескольким клиентам тестирования, а клиент тестирования может быть подключен только к одному менеджеру.

Для управления клиентом менеджер устанавливает с ним *TCP*-соединение. Важно, что для проведения автоматизированного тестирования не требуется вносить изменений в структуру конфигурации.

По сути, клиент и менеджер тестирования – это конфигурации, запущенные с определенными параметрами командной строки, причем менеджер осуществляет управление клиентами, “заставляя” окна и элементы управления вести себя таким образом, как будто с ними взаимодействует пользователь.

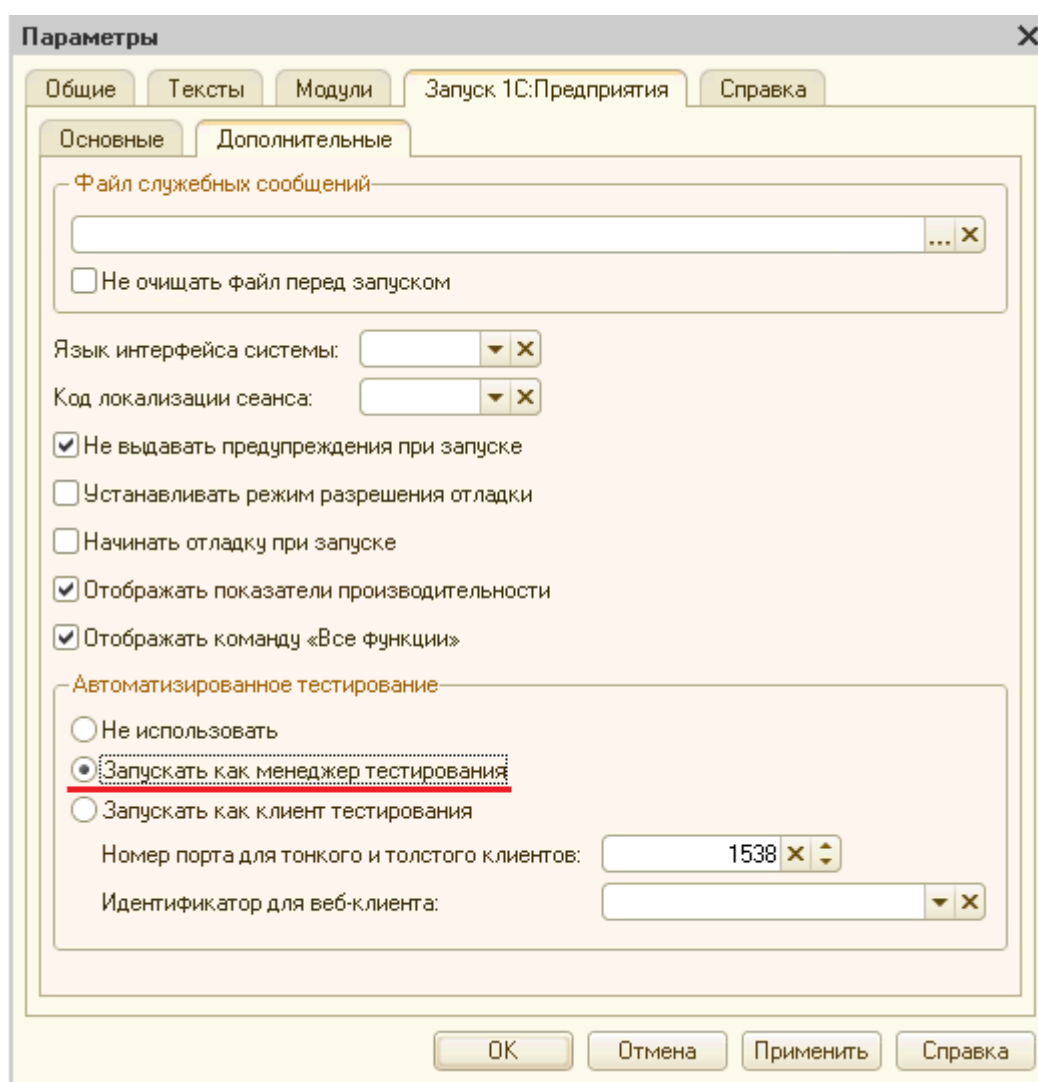
Для выполнения автоматизированного тестирования должен быть запущен менеджер и клиент тестирования.

Запуск менеджера может быть выполнен из командной строки с ключом */TESTMANAGER*:

`"c:\Program Files (x86)\1cv8\8.3.4.437\bin\1cv8c.exe" ENTERPRISE /F "X:\test" /N Администратор /TESTMANAGER`

Также менеджер тестирования можно запустить из конфигуратора.

Для этого через меню *Сервис - Параметры* открываем диалог *"Параметры"*, в котором на закладке *Запуск 1С:Предприятия - Дополнительные* отмечаем пункт *"Запустить как менеджер тестирования"*:



Еще один способ запуска менеджера тестирования – из встроенного языка, при помощи метода *ЗапуститьСистему()*, в котором следует указать командную строку:

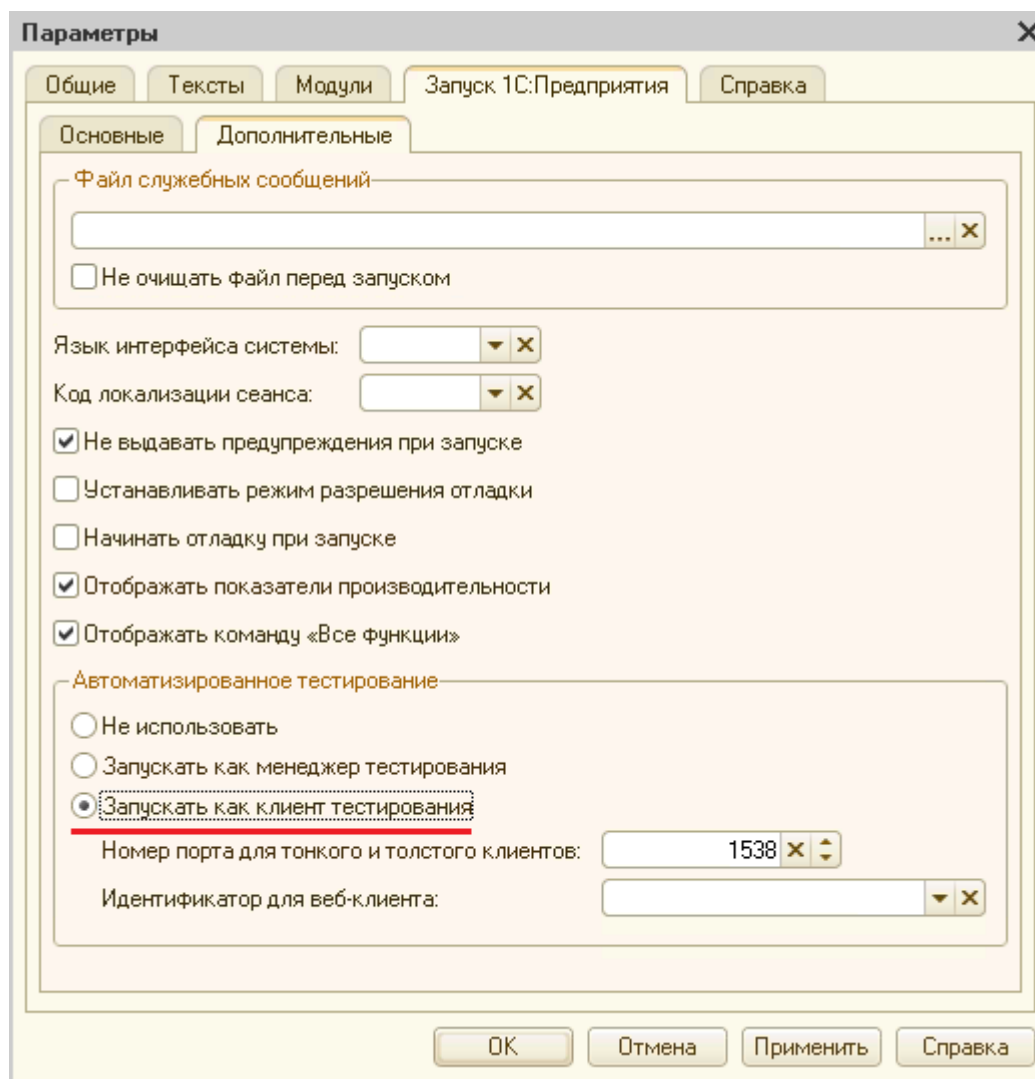
```
ЗапуститьСистему("c:\Program Files (x86)\1cv8\8.3.4.437\bin\1cv8c.exe" ENTERPRISE /F X:\test /N Администратор /TESTMANAGER")
```

Клиента тестирования также можно запустить из командной строки. Для этого следует воспользоваться ключом параметра командной строки */TESTCLIENT*.

При помощи параметра *TPort* указывается номер порта, через который будет вестись взаимодействие менеджера и клиента тестирования. Если этот параметр не указан в командной строке, то будет использован порт 1538.

```
"c:\Program Files (x86)\1cv8\8.3.4.437\bin\1cv8c.exe" ENTERPRISE /F "X:\Platform8Demo" /N Администратор /TESTCLIENT -TPort 1539
```

Клиент тестирования можно запустить из конфигуратора. Для этого через меню *Сервис - Параметры* открываем диалог *"Параметры"*, в котором на закладке *Запуск 1С:Предприятия - Дополнительные* отмечаем пункт *"Запускать как клиент тестирования"*. При этом надо будет указать номер используемого порта.



Обратите внимание, что для подключения к клиенту тестирования необходимо знать два параметра: IP-адрес (или имя) компьютера, на котором запущен клиент тестирования, и номер TCP-порта, с помощью которого будет выполняться взаимодействие.

В качестве менеджера и клиента тестирования могут использоваться как разные информационные базы (конфигурация базы менеджера тестирования может не совпадать с конфигурацией клиента тестирования), так и одна и та же информационная база.

Для выполнения автоматизированного тестирования необходимо проделать следующие шаги:

1. Разработать сценарий тестирования – написать внешнюю или встроенную в

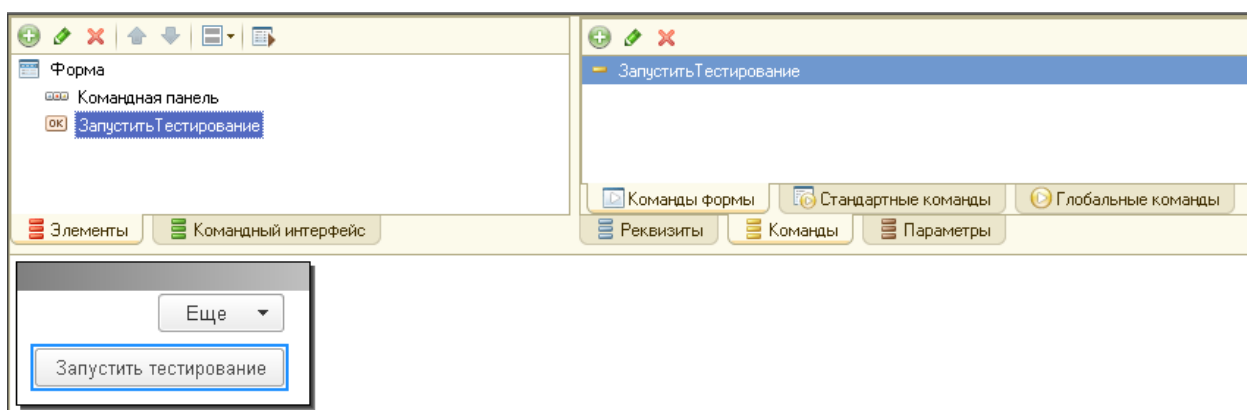
конфигурацию обработку, в которой будут последовательно описаны выполняемые шаги.

2. Запустить менеджер тестирования.
3. Запустить клиент тестирования (один или несколько).
4. В менеджере тестирования запустить на исполнение созданную обработку, убедиться в выполнении запрограммированных действий на клиенте.

Тестируемое приложение описывается набором объектов встроенного языка, которые используются для написания сценария:

- ТестируемоеПриложение;
- ТестируемоеОкноКлиентскогоПриложения;
- ТестируемыйКомандныйИнтерфейсОкна;
- ТестируемаяГруппаКомандногоИнтерфейса;
- ТестируемаяКнопкаКомандногоИнтерфейса;
- ТестируемаяФорма;
- ТестируемоеПолеФормы;
- ТестируемаяГруппаФормы;
- ТестируемаяКнопкаФормы;
- ТестируемаяТаблицаФормы;
- ТестируемаяДекорацияФормы.

В качестве тестируемой конфигурации будем использовать демонстрационную конфигурацию "Управляемое приложение". Создадим внешнюю обработку, добавим новую форму, в которой определим обработчик для новой команды "ЗапуститьТестирование".



В тесте выполняем следующие действия: создаем новый элемент справочника

“Склады”, в поле *Наименование* вводим строку “Склад тест”, затем нажимаем кнопку “Записать и закрыть”. Программный код этого теста будет выглядеть следующим образом:

```
&НаКлиенте
Процедура ЗапуститьТестирование (Команда)
    // Подключимся к тестируемому приложению
    ТестируемоеПриложение = Новый ТестируемоеПриложение ("localhost");
    // Пытаемся подключаться не более одной минуты
    ВремяОкончанияОжидания = ТекущаяДата () + 60;
    Подключен = Ложь;
    Пока Не ТекущаяДата () >= ВремяОкончанияОжидания Цикл

        Попытка
            ТестируемоеПриложение.УстановитьСоединение ();
            Подключен = Истина;
            Прервать;
        Исключение
            КонецПопытки;

    КонецЦикла;

    Если Не Подключен Тогда
        // Завершаем работу теста
        ТестируемоеПриложение = Неопределено;
        Сообщить ("Не удалось установить соединение!");
        Возврат;
    КонецЕсли;
    // Найдем главное окно
    ГлавноеОкноТестируемого =
    ТестируемоеПриложение.НайтиОбъект (Тип ("ТестируемоеОкноКлиентскогоПриложения")
);
    ГлавноеОкноТестируемого.Активизировать ();
    // Выполним команду создания элемента справочника товаров

    ГлавноеОкноТестируемого.ВыполнитьКоманду ("elcib/command/Справочник.Склады.Создать");
    ТестируемоеПриложение.ОжидатьОтображениеОбъекта (Тип ("ТестируемаяФорма"),
"Склад*");
    ТестируемаяФорма =
    ТестируемоеПриложение.НайтиОбъект (Тип ("ТестируемаяФорма"), "Склад*");
    ТестируемаяФорма.Активизировать ();
    // Зададим наименование для нового товара
    ЭлементФормы = ТестируемаяФорма.НайтиОбъект (Тип ("ТестируемоеПолеФормы"),
"Наименование");
    ЭлементФормы.Активизировать ();
    ЭлементФормы.ВвестиТекст ("Склад тест");
    // Запишем элемент
    ЭлементФормы =
    ТестируемаяФорма.НайтиОбъект (Тип ("ТестируемаяКнопкаФормы"), "Записать и
закрыть");
    ЭлементФормы.Нажать ();
КонецПроцедуры
```

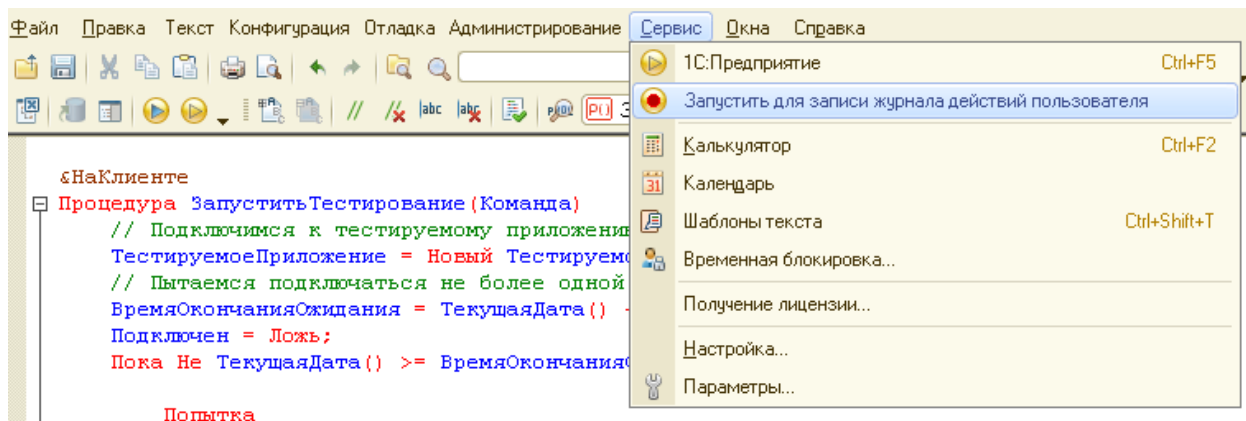
В диалоге параметров запуска сначала выбиралось значение *“Запустить как менеджер тестирования”*, при помощи сочетания клавиш *Ctrl+F5* запускался пользовательский сеанс.

Потом в диалоге выбиралось значение *“Запустить как клиент тестирования”*, при помощи сочетания клавиш *Ctrl+F5* запускался второй пользовательский сеанс.

Таким образом мы избежали необходимости вручную указывать требуемые параметры командной строки.

Приведенный выше код выполняет довольно простые действия, но в случае усложнения сценария возрастет и объем кода, поскольку необходимо описать каждое интерактивное действие пользователя.

Здесь на помощь приходит еще одна новая возможность платформы – запись журнала действий пользователя. Для этого необходимо запустить приложение в специальном режиме:



В заголовке программы появляется несколько кнопок:

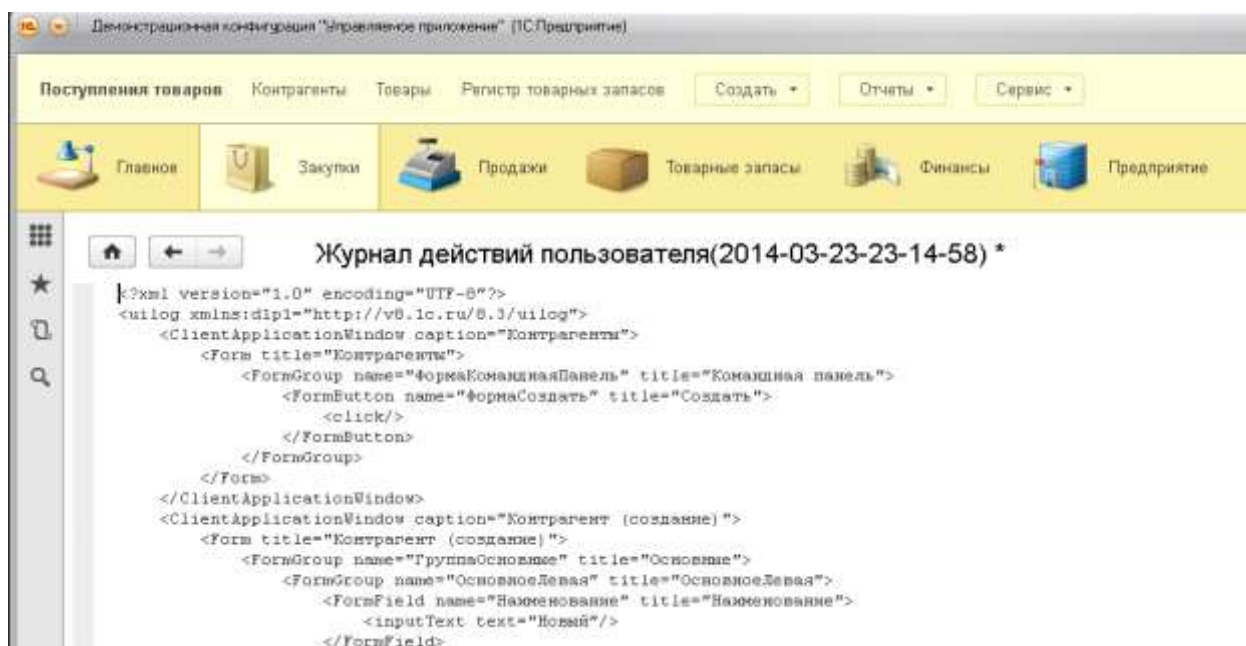


Кнопки предназначены для:

- начала/приостановки записи;
- прекращения записи;
- завершения записи.

После завершения записи на экране открывается текстовый документ, который

представляет собой последовательность действий пользователя, сохраненную в XML-файл.



Записанный журнал можно преобразовать в программный код, который затем использовать в сценарии тестирования. Для этого предназначена обработка *“Преобразование журнала действий пользователя” (UILogToScript.epf)*, которую можно получить с [сайта ИТС](#).



## Преобразование журнала действий пользователя

► Преобразовать

### Параметры преобразования

Генерировать сценарий в варианте встроенного языка:	<input type="text" value="Русский"/>
Имя основной процедуры:	<input type="text" value="ТестовыйСценарий_23_03_2014"/>
Генерировать код подключения к клиенту:	<input checked="" type="checkbox"/>
Разбивать результат на процедуры по "границам" окон:	<input checked="" type="checkbox"/>
Генерировать получение родительских объектов в иерархии:	<input type="checkbox"/>
Поиск объектов:	<input checked="" type="radio"/> По представлению <input type="radio"/> По имени <input type="radio"/> По имени и представлению

Преобразовывать:  Журнал  Текст

В результате работы обработки мы получаем сгенерированный код на встроенном языке. Этот код следует вставить в модуль формы обработки тестирования.



Обратите внимание, что в сгенерированном коде числа, большие 999 или меньшие -999, будут выводиться с использованием неразрывного пробела в качестве разделителя групп (например, «1 234» вместо «1234»).

Этот символ необходимо удалить из полученного кода вручную. Участок кода с подключением к клиенту обработка сформировала автоматически.

В нашем примере получился следующий код:

```
&НаКлиенте
Процедура ЗапуститьТестирование (Команда)
    ТестовыйСценарий_23_03_2014 ();
КонецПроцедуры

&НаКлиенте
Процедура ТестовыйСценарий_23_03_2014 ()

    ТестовоеПриложение = Новый ТестируемоеПриложение ();
    ВремяОкончанияОжидания = ТекущаяДата () + 60;
    Подключен = Ложь;
    ОписаниеОшибкиСоединения = "";
    Пока Не ТекущаяДата () >= ВремяОкончанияОжидания Цикл
        Попытка
            ТестовоеПриложение.УстановитьСоединение ();
            Подключен = Истина;
            Прервать;
        Исключение
            ОписаниеОшибкиСоединения = ОписаниеОшибки ();
        КонецПопытки;
    КонецЦикла;
    Если Не Подключен Тогда
        ТестовоеПриложение = Неопределено;
        Сообщить ("Не смогли установить соединение! " + Символы.ПС +
        ОписаниеОшибкиСоединения);
    Возврат;
    КонецЕсли;

    ОкноПриложенияКонтрагентыКнопкаСоздатьНажать (ТестовоеПриложение);

    ОкноПриложенияКонтрагентСозданиеКнопкаЗаписатьИЗакретьНажать (ТестовоеПриложение);

КонецПроцедуры

&НаКлиенте
Процедура ОкноПриложенияКонтрагентыКнопкаСоздатьНажать (ТестовоеПриложение)
    ОкноПриложенияКонтрагенты =
    ТестовоеПриложение.НайтиОбъект (Тип ("ТестируемоеОкноКлиентскогоПриложения"),
    "Контрагенты", , 30);
    ОкноПриложенияКонтрагентыФормаКонтрагенты =
    ОкноПриложенияКонтрагенты.НайтиОбъект (Тип ("ТестируемаяФорма"),
    "Контрагенты");
```

```
КнопкаСоздать =  
ОкноПриложенияКонтрагентыФормаКонтрагенты.НайтиОбъект (Тип ("ТестируемаяКнопкаФормы"), "Создать");  
КнопкаСоздать.Нажать ();
```

## КонецПроцедуры

&НаКлиенте

## Процедура

```
ОкноПриложенияКонтрагентСозданиеКнопкаЗаписатьИЗакрытьНажать (ТестовоеПриложение)
```

```
ОкноПриложенияКонтрагентСоздание =  
ТестовоеПриложение.НайтиОбъект (Тип ("ТестируемоеОкноКлиентскогоПриложения"),  
"Контрагент (создание)", , 30);
```

```
ОкноПриложенияКонтрагентСозданиеФормаКонтрагентСоздание =  
ОкноПриложенияКонтрагентСоздание.НайтиОбъект (Тип ("ТестируемаяФорма"),  
"Контрагент (создание)");
```

```
ПолеНаименование =  
ОкноПриложенияКонтрагентСозданиеФормаКонтрагентСоздание.НайтиОбъект (Тип ("ТестируемоеПолеФормы"),  
"Наименование");  
ПолеНаименование.ВвестиТекст ("Новый");
```

```
ПолеВидЦен =  
ОкноПриложенияКонтрагентСозданиеФормаКонтрагентСоздание.НайтиОбъект (Тип ("ТестируемоеПолеФормы"),  
"Вид цен");
```

```
ПолеВидЦен.Активизировать ();
```

```
ПолеВидЦен.Выбрать ();
```

```
ПолеВидЦен.ОжидатьФормированияВыпадающегоСписка ();
```

```
ПолеВидЦен.ВыполнитьВыборИзСпискаВыбора ("Закупочная");
```

```
КнопкаЗаписатьИЗакрыть =  
ОкноПриложенияКонтрагентСозданиеФормаКонтрагентСоздание.НайтиОбъект (Тип ("ТестируемаяКнопкаФормы"),  
"Записать и закрыть");  
КнопкаЗаписатьИЗакрыть.Нажать ();
```

## КонецПроцедуры

В полученном сценарии устанавливается подключение к клиенту тестирования, нажимается кнопка создания нового элемента справочника “Контрагенты”, в поле *Наименование* вводится текст “Новый”, а в выпадающем списке “Вид цен” выбираем значение “Закупочная”, затем нажимается кнопка “Записать и закрыть”.

Если в сценарии необходимо использовать несколько клиентов тестирования, то подключение к каждому из них и выполняемые действия необходимо описать отдельно.

Менеджер тестирования будет использоваться один, а к нему подключены два

клиента на разных портах.

Поскольку сценарий тестирования – это обработка, строки программного кода которой выполняются последовательно, то разработчику необходимо описать последовательность действий для каждого клиента.

Рассмотрим подробнее, как будет выглядеть код при использовании двух клиентов тестирования:

**Процедура** ТестовыйСценарий\_23\_03\_2014\_ДваКлиента ()

```
//первый клиент
ТестовоеПриложение1 = Новый ТестируемоеПриложение ();
ВремяОкончанияОжидания = ТекущаяДата () + 60;
Подключен = Ложь;
ОписаниеОшибкиСоединения = "";
Пока Не ТекущаяДата () >= ВремяОкончанияОжидания Цикл
    Попытка
        ТестовоеПриложение1.УстановитьСоединение ();
        Подключен = Истина;
        Прервать;
    Исключение
        ОписаниеОшибкиСоединения = ОписаниеОшибки ();
    КонечПопытки;
КонечЦикла;

//второй клиент
ТестовоеПриложение2 = Новый ТестируемоеПриложение ();
ВремяОкончанияОжидания = ТекущаяДата () + 60;
ОписаниеОшибкиСоединения = "";
Пока Не ТекущаяДата () >= ВремяОкончанияОжидания Цикл
    Попытка
        ТестовоеПриложение2.УстановитьСоединение ();
        Подключен = Истина;
        Прервать;
    Исключение
        ОписаниеОшибкиСоединения = ОписаниеОшибки ();
    КонечПопытки;
КонечЦикла;
Если Не Подключен Тогда
    ТестовоеПриложение1 = Неопределено;
    ТестовоеПриложение2 = Неопределено;
    Сообщить ("Не смогли установить соединение! " + Символы.ПС +
ОписаниеОшибкиСоединения);
    Возврат;
КонечЕсли;

//процедуры отдельные для каждого клиента тестирования
ОкноПриложенияКонтрагентыКнопкаСоздатьНажать1 (ТестовоеПриложение1);
ОкноПриложенияКонтрагентыКнопкаСоздатьНажать2 (ТестовоеПриложение2);
```

[ОкноПриложенияКонтрагентСозданиеКнопкаЗаписатьИЗакрытьНажать1 \(ТестовоеПриложение1\) ;](#)

[ОкноПриложенияКонтрагентСозданиеКнопкаЗаписатьИЗакрытьНажать2 \(ТестовоеПриложение2\) ;](#)

## **КонецПроцедуры**

Паузы между выполняемыми действиями тоже нужно запрограммировать отдельно. Сценарий для большого количества клиентов становится трудночитаемым. Кроме того, автоматизированное тестирование доступно только для управляемых форм.

Однако преимуществом автоматизированного тестирования является простота и наглядность разработки тестов.

Поскольку тест оперирует только интерактивными действиями пользователя, то разработчику не нужно знать структуры конфигурации на уровне реквизитов объектов.

При изменении, например, кода конфигурации нет необходимости переделывать тест, поскольку на клиенте тестирования по-прежнему будут выполняться те же самые действия с теми же самыми элементами управления.

Механизм автоматизированного тестирования может быть использовано тестировщиками для записи последовательности действий, приводящих к ошибке.

Записанные данные можно отправить разработчикам для исправления обнаруженной ошибки.

Также автоматизированное тестирование может применяться для выявления в конфигурации избыточных блокировок и взаимоблокировок. Для этого нужно реализовать сценарий, воспроизводящий проблему, и приступить к поиску и устранению причины.

**Ханевич Василий**

г. Калининград

## Дополнительные материалы

Все статьи проекта Курсы-по-1С.рф: <http://курсы-по-1с.рф/blog/articles/>

## Курсы по программированию в 1С v.8

Базовый и Продвинутой курсы по Программированию на Платформе 1С 8  
<http://www.Spec8.ru/>



### Базовый курс по программированию в 1С v.8

Курс про **готовые приемы и решения**  
**90% задач** по программированию в 1С



### Продвинутой курс по программированию в 1С v.8

Больше, чем Вы можете себе представить  
Детальнее требований на **1С:Специалист**

«Курс по подготовке к Аттестации по Платформе 1С 8.2 / 8.3»

<http://курсы-по-1с.рф/dev-attestation/>



### Подготовка к Аттестации по Платформе 1С v.8

Аттестация по Платформе – **с первого раза**  
**Экономия 100 - 150 часов** подготовки