

Getting started with OpenCTF

Michael Justin

A short guide for the first steps with the Open Source Component Test Framework

Trademarks

All Embarcadero brands and product names are trademarks or registered trademarks of Embarcadero. Oracle and Java are registered trademarks of Oracle and/or its affiliates. Microsoft, Windows, Windows NT, and/or other Microsoft products referenced herein are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. Other brands and their products are trademarks of their respective holders.

Introduction

About OpenCTF

How can I use it?

The OpenCTF component test framework helps to build automatic tests for all (visual and non-visual) VCL components in a Delphi application. It is based on the DUnit framework.

How does it work?

- OpenCTF iterates over all specified Forms and DataModules and dynamically creates DUnit test cases for selected components
- Customized test classes detect which components have to be tested, and configure the test steps

How can it help me?

OpenCTF provides an easy way to build automatic quality checks for large projects where many components have to pass repeated tests.

Here are some examples:

- detect missing or wrong property values - e.g. Buttons without assigned Actions, DataSources without associated DataSet
- detect unassigned event handlers - e.g. missing OnExecute event
- check that all DataSets can be opened
- check the tab order
- find invisible components (e.g. invisible TabSheets which better should be hidden at runtime)

Is it easy to use?

Creating and customizing tests for your specific needs is very easy:

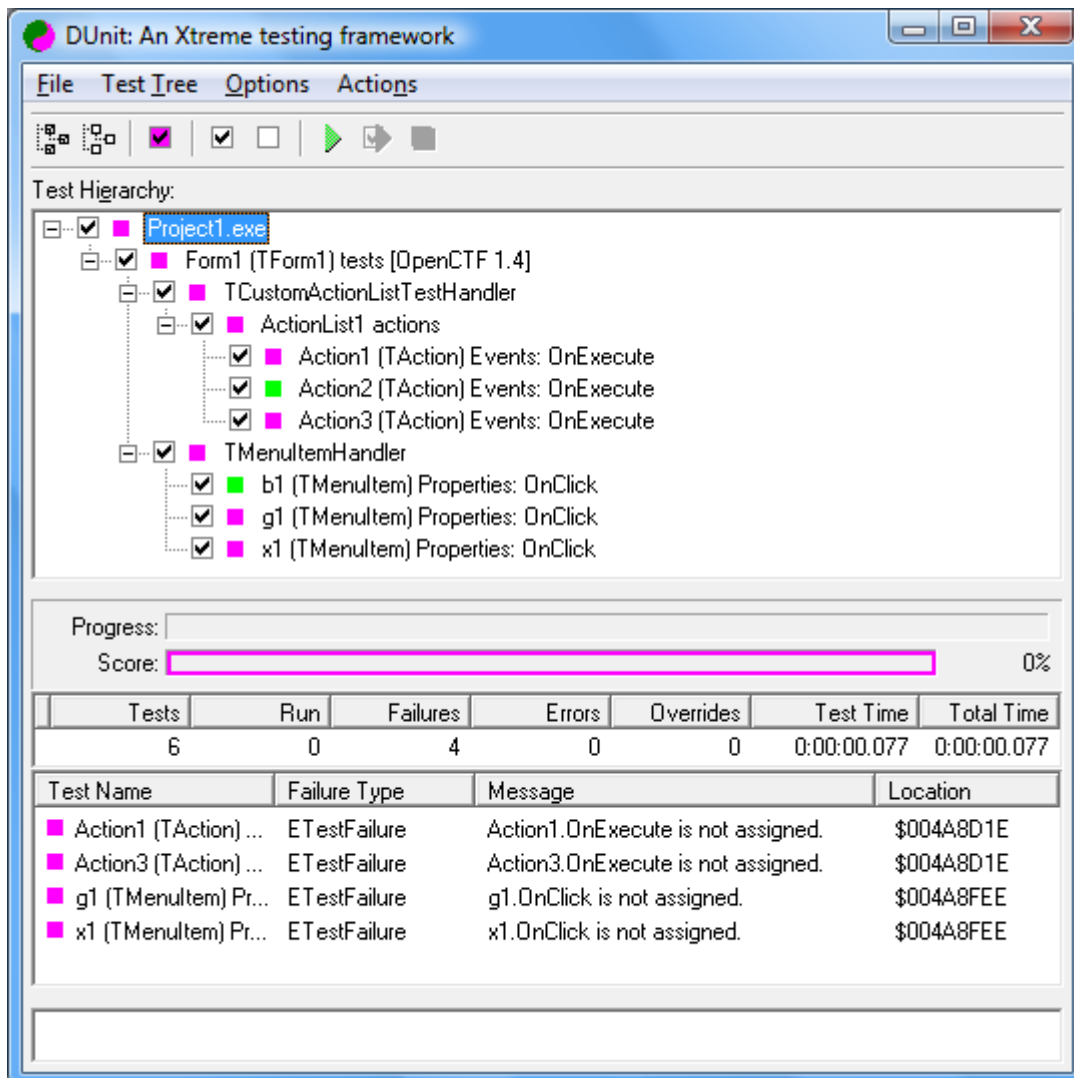
- Adding OpenCTF tests to a DUnit test suite requires only one or two lines of code
- Writing your own custom component tests is simple, many examples are included in the source distribution

Hosted on Sourceforge

- OpenCTF is hosted on Sourceforge:
<http://sourceforge.net/projects/openctf/>
- API documentation is available online at <http://openctf.sourceforge.net/>

Key features:

- easy to use and extend
- includes example tests (including simple test for ADO, dbExpress, IBX) and example projects
- contains the full source code
- based on [DUnit](#)



Example Test Program

Simple Example

```
program Program1;

{$APPTYPE CONSOLE}

uses
  OpenCTF,
  OpenCTFRunner,
  ctfTestActnList,
  ctfTestControls,
  ctfTestMenus,
  TestForm in 'TestForm.pas' {Form1};

begin
  // Register Form classes
  OpenCTF.RegisterFormClasses([TForm1]);
  // run the tests
  OpenCTFRunner.Run;
end.
```

In this example, the framework will run all registered tests on an instance of TForm1.