Database Unit Testing With TSQLT

Scofield

Content

- ❑ Why Unit Testing SQL?
- ❑ tSQLt Framework
- ❑ Scenarios
- ❑ Limitations
- ❑ Unit Testing Integration

Why Unit Testing SQL?

- Enforce good design
- Noticed broken things early

A good unit test

- Able to be fully **automated** and **repeatable**
- **Consistently** returns the same result
- **Independent** (Use mocks or stubs to achieve this isolation when needed)
- Runs **fast**
- Tests a **single logical concept** in the system
- **Readable**
- **Maintainable**
- **Trustworthy** (when you see its result, you don't need to debug the code just to be sure)

tSQLt Framework

http://tsqlt.org

- Open Source
- Written in tsqlt and C#
- Runs on database level
- Require CLR (SQL Server 2005 SP2+)

Example

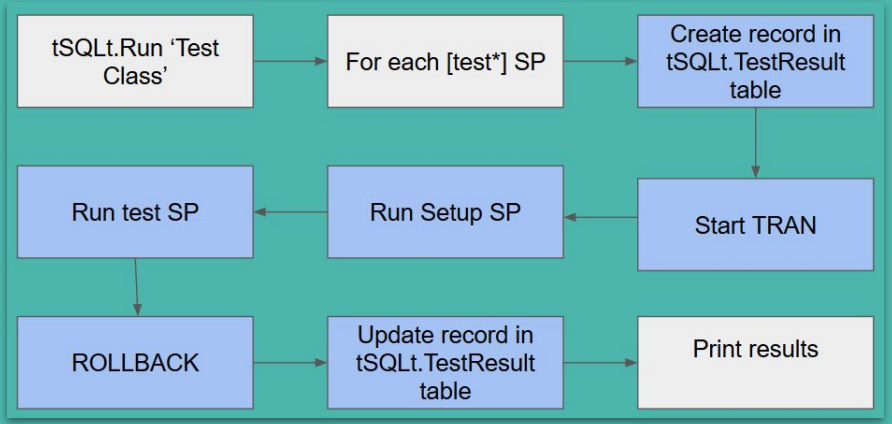
```
ALTER PROCEDURE [dbo].[OIC_LanguageSel]
AS

/*
    Created by: Dom
    Date: 2014-12-18
    Task: Get Language list [Redmine:#27752]
    DB: DBOIC.bodb_OIC

    Revisions:
        - 20160826@Percy: Remove Other Language (ID = 4) [RedmineID: #62883]

 */

BEGIN
    SET NOCOUNT ON
    SELECT LanguageId
        ,LanguageName
    FROM dbo.[Language] WITH(NOLOCK)
    WHERE LanguageId <> 4
    ORDER BY LanguageName
END
```

We want to test that this SP will return result with LanguageId != 4

The first test

```
ALTER PROCEDURE [UnitTest].[Test OIC_LanguageSel: When Successful - Return Language Without LanguageId = 4 ]
AS
BEGIN
   -- Arrange
    CREATE TABLE #Actual (
       LanguageID      INT,
       LanguageeName       VARCHAR(30)
    )

    DECLARE @Count INT

    -- Act
    INSERT #Actual
    EXEC [dbo].[OIC_LanguageSel]

    SELECT @Count = Count(*)
    FROM #Actual
    WHERE LanguageID = 4

    -- Assert
    EXEC tSQLt.AssertEquals  0, @Count
END;
```

tSQLt Framework



tSQLt.Run

Test Creation

```
tSQLt.NewTestClass 'UnitTest'
```

Test Execution

```
tSQLt.Run '[Schema].[Test Name]'
```

```
tSQLt.RunAll
```

Assertions

- AssertEquals / AssertNotEquals

```
EXEC tSQLt.AssertEquals 12345.6789, 54321.123; -- fail
```

```
EXEC tSQLt.AssertEquals NULL, NULL; -- pass
```

- AssertEqualsString

```
EXEC tSQLt.AssertEqualsString 'hello', N'hello';                    --
pass (values are compared as NVARCHAR(MAX))
```

- AssertLike

```
EXEC tSQLt.AssertLike '%el%', 'hello'; -- pass
```

Assertions

- AssertEmptyTable

```
EXEC tSQLt.AssertEmptyTable 'actual';
```

- AssertEqualsTable

```
EXEC tSQLt.AssertEqualsTable 'expected', 'actual';
```

- AssertEqualsTableSchema / AssertResultSetsHaveSameMetaData

```
EXEC tSQLt.AssertEqualsTableSchema 'mySchema.ThisTable',
'mySchema.ThatTable';
```

```
EXEC tSQLt.AssertResultSetsHaveSameMetaData
'SELECT TOP 1 * FROM mySchema.ThisTable;',
'SELECT TOP 1 * FROM mySchema.ThatTable;';
```

Assertions

- AssertObjectExist

```
EXEC tSQLt.AssertObjectExists 'UpdateTable';
```

- AssertObjectDoesNotExist

```
EXEC tSQLt.AssertObjectDoesNotExists 'dbo.MyProcedure';
```

Assertions

- Fail

```
EXEC tSQLt.Fail 'An error message'
```

Isolating Dependencies

- **FakeTable**
- **ApplyConstraint**
- **FakeFunction**
- **SpyProcedure**
- RemoveObject
- RemoveObjectIfExists
- ApplyTrigger

FakeTable

```
tSQLt.FakeTable 'table name'
```

- Create an empty version of table without constraints
- Cannot be used with temporary tables

FakeFunction

```
tSQLt.FakeFunction 'function name', 'fake function name'
```

- To isolate the code we are testing from the logic buried in the functions that it calls
- Both functions must be compatible in function types and parameters
- A real function needs to be created

SpyProcedure

```
tSQLt.SpyProcedure 'procedure name'
```

- Allows tests to be written in isolation of the other procedures that it calls
- Creates a table @ProcedureName + '_SpyProcedureLog' containing procedure parameters
- Can not be used with temporary stored procedures (whose name begins with #)

ApplyConstraint

```
tSQLt.ApplyConstraint 'table name', 'constraint name'
```

- Allows constraints to be tested in isolation of other constraints on a table
- Works with the following constraint types:
  - CHECK constraints
  - FOREIGN KEY constraints
  - UNIQUE constraints
  - PRIMARY KEY constraints

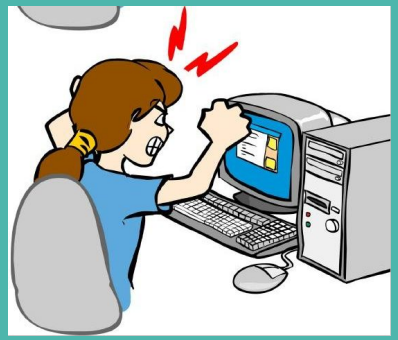
How FakeTable works?

```
tSQLt.FakeTable 'Language'
```

```
SELECT
        OriginalName,
        SCHEMA_NAME(schema_id) + '.' + name as [Renamed Table],
        create_date
FROM tSQLt.Private_RenamedObjectLog
        JOIN sys.objects
        ON objectid = object_id
```

OriginalName	Name of Renamed Table	create_date
[Language]	dbo.tSQLt_tempobject_6567ba995e024dce93103f7531e28057	2018-04-09 13:26:52.890

Side affects!



Demos

- FakeTable
- Get Output Params
- Test Constraint
- Fake Function
- Spy Procedure
- Multiple Result Sets
- Fake Table From a Different Database

Limitations

- Fake tables create columns as nullable, even when they were non-nullable
- Isolating Dependencies can not work with temporary tables, stored procedures
- ApplyConstraint limits

Limitations

- INSERT EXEC Nested Problem

Limitations

- Watch out for time (cannot run tests in parallel)
  - First run

No	Test Case Name	Dur(ms)	Result
1	[UnitTest].[Test Constraint: LicenseeType_LicenseeTypeName_Unique]	586	Success
2	[UnitTest].[Test OIC_LanguageSel: When Successful - Return Language Without LanguageId = 4]	6	Success
3	[UnitTest].[Test OIC_LicenseeTypeSel: When Successful - Return Right LicenseeType]	30	Success
4	[UnitTest].[Test OIC_LicenseeTypeUpd: When LicenseeTypeName Not Exist - Update Successful]	13	Success
5	[UnitTest].[Test OIC_LicenseeTypeUpd: When Update Successful - Write Log]	40	Success
6	[UnitTest].[Test OIC_LicenseeTypeUpd: When LicenseeTypeName Exist Return Errors]	13	Error

  - Second run

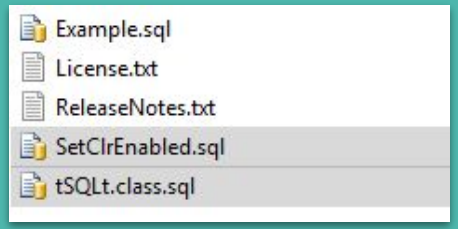
No	Test Case Name	Dur(ms)	Result
1	[UnitTest].[Test Constraint: LicenseeType_LicenseeTypeName_Unique]	10	Success
2	[UnitTest].[Test OIC_LanguageSel: When Successful - Return Language Without LanguageId = 4]	6	Success
3	[UnitTest].[Test OIC_LicenseeTypeSel: When Successful - Return Right LicenseeType]	40	Success
4	[UnitTest].[Test OIC_LicenseeTypeUpd: When LicenseeTypeName Not Exist - Update Successful]	10	Success
5	[UnitTest].[Test OIC_LicenseeTypeUpd: When Update Successful - Write Log]	36	Success
6	[UnitTest].[Test OIC_LicenseeTypeUpd: When LicenseeTypeName Exist Return Errors]	3	Error

Integration

- Setup seperate databases
- Integrate to Jenkins

Integration

- https://tsqlt.org/downloads/



- Run scripts:
  - SetClrEnable.sql
  - tSQLt.class.sql

Integration

- Test Result

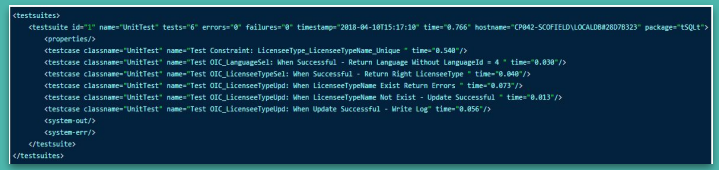
```
SELECT * FROM tSQLt.TestResult
```

	Id	Class	TestCase	Name	TranName	Result	Msg	TestStartTime	TestEndTime
1	1142	UnitTest	Test Constraint: LicenseeType_LicenseeTypeName_Unique	[UnitTest].[Test Constraint: LicenseeType_LicenseeT...	tSQLtTran1E1BC7052C034C34B741C8E	Success		2018-04-10 15:20:36.393	2018-04-10 15:20:36.940
2	1143	UnitTest	Test OIC_LanguageSel: When Successful - Return Langu...	[UnitTest].[Test OIC_LanguageSel: When Successful...	tSQLtTran329E8466BC5A466A956670A0	Success		2018-04-10 15:20:36.943	2018-04-10 15:20:36.957
3	1144	UnitTest	Test OIC_LicenseeTypeSel: When Successful - Return Ri...	[UnitTest].[Test OIC_LicenseeTypeSel: When Succe...	tSQLtTranA7586F0EF56D4698A22FF90	Success		2018-04-10 15:20:36.960	2018-04-10 15:20:36.987
4	1145	UnitTest	Test OIC_LicenseeTypeUpd: When LicenseeTypeName ...	[UnitTest].[Test OIC_LicenseeTypeUpd: When Licen...	tSQLtTran4B7040A2C3664D188E1F716	Success		2018-04-10 15:20:36.993	2018-04-10 15:20:37.057
5	1146	UnitTest	Test OIC_LicenseeTypeUpd: When LicenseeTypeName ...	[UnitTest].[Test OIC_LicenseeTypeUpd: When Licen...	tSQLtTran4422B189A58F42BF9A4C7535	Success		2018-04-10 15:20:37.060	2018-04-10 15:20:37.070
6	1147	UnitTest	Test OIC_LicenseeTypeUpd: When Update Successful - ...	[UnitTest].[Test OIC_LicenseeTypeUpd: When Updat...	tSQLtTran09AA8BDCE9524534BF667B8	Success		2018-04-10 15:20:37.077	2018-04-10 15:20:37.120

Integration

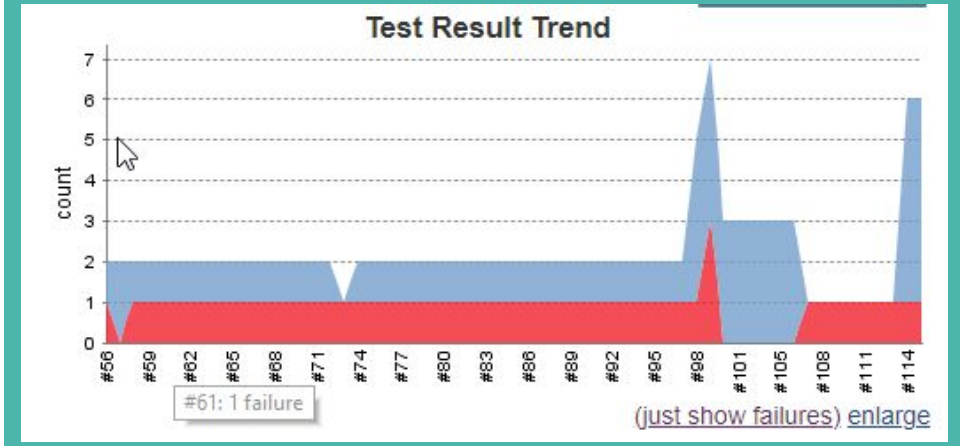
- Test Result

```
EXEC  tSQLt.XmlResultFormatter
```



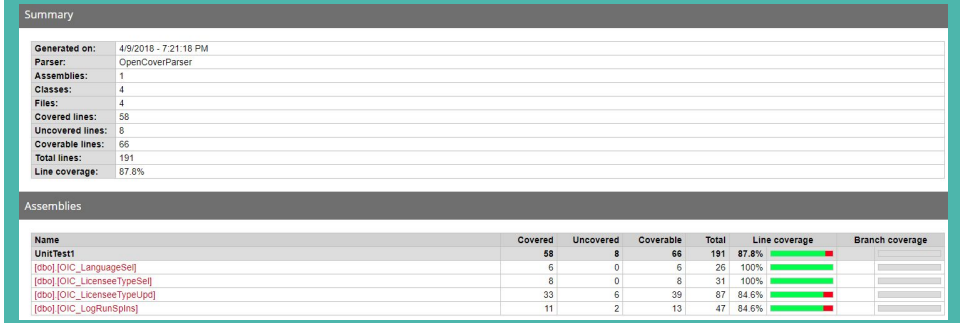
```
<testsuites>
   <testsuite id="1" name="UnitTest" tests="6" errors="0" failures="0" timestamp="2018-04-10T15:17:10" time="0.766" hostname="CP042-SCOFIELD\LOCALDB#2BD7B323" package="tSQLt">
      <properties/>
      <testcase classname="UnitTest" name="Test Constraint: LicenseeType_LicenseeTypeName_Unique " time="0.540"/>
      <testcase classname="UnitTest" name="Test OIC_LanguageSel: When Successful - Return Language Without LanguageId = 4 " time="0.030"/>
      <testcase classname="UnitTest" name="Test OIC_LicenseeTypeSel: When SuccessFul - Return Right LicenseeType " time="0.040"/>
      <testcase classname="UnitTest" name="Test OIC_LicenseeTypeUpd: When LicenseeTypeName Exist Return Errors " time="0.073"/>
      <testcase classname="UnitTest" name="Test OIC_LicenseeTypeUpd: When LicenseeTypeName Not Exist - Update Successful " time="0.013"/>
      <testcase classname="UnitTest" name="Test OIC_LicenseeTypeUpd: When Update Successful - Write Log" time="0.056"/>
      <system-out/>
      <system-err/>
   </testsuite>
</testsuites>
```

Jenkin Integration



tSQLt.XmlResultFormatter + JUnit

Jenkin Integration



Summary

Generated on:	4/9/2018 - 7:21:18 PM
Parser:	OpenCoverParser
Assemblies:	1
Classes:	4
Files:	4
Covered lines:	58
Uncovered lines:	8
Coverable lines:	66
Total lines:	191
Line coverage:	87.8%

Assemblies

Name	Covered	Uncovered	Coverable	Total	Line coverage	Branch coverage
UnitTest1	58	8	66	191	87.8%	
[dbo].[OIC_LanguageSel]	6	0	6	26	100%	
[dbo].[OIC_LicenseeTypeSel]	8	0	8	31	100%	
[dbo].[OIC_LicenseeTypeUpd]	33	6	39	87	84.6%	
[dbo].[OIC_LogRunSpIns]	11	2	13	47	84.6%	

SQLCover + Report Generator

Thank You!