

FUTURE DECODED

6-7 OTT '16 / MILANO

IN PARTNERSHIP WITH:



CommunityDays.it

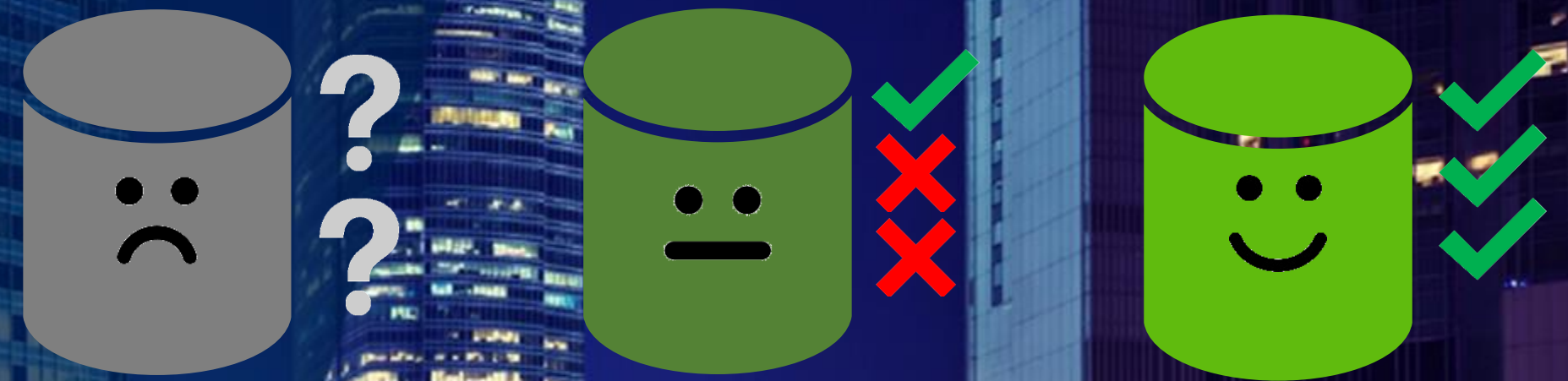
www.futuredecoded.it



#FutureDecoded

Unit Testing con SQL Server

Alessandro Alpi
@suxstellino



```
4 SET ANSI_NULLS ON
5 GO
6 SET QUOTED_IDENTIFIER ON
7 GO
8 ALTER PROCEDURE [UnitTests].[test this speech should be useful]
9 AS
10 BEGIN
11
12 DECLARE @UserWalletId int;
```

www.futuredecoded.it

 #FutureDecoded



Alessandro Alpi

Microsoft MVP – SQL Server dal 2008

Blog ITA: <http://blogs.dotnethell.it/suxstellino>

Blog ENG: <http://suxstellino.wordpress.com/>

CTO Engage IT Services S.r.l.

www.engageitservices.it

Team leader (Agile)

Communities

Getlatestversion.it



Microsoft
CERTIFIED
IT Professional

Microsoft
CERTIFIED
Professional



Agenda

Concetti di Unit Testing

Perché Unit Testing su database

Unit Testing framework

Unit Testing solution

Conclusioni

Q&A

Concetto di unit test

*In computer programming, unit testing is a software testing method by which **individual units of source code**, sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures are tested to determine if they **are fit for use**. The primary purpose of this approach is **to find out bugs and prevent regressions**.*

Parleremo di..

Development teams

Codice e funzionalità

QA Team

Evitare e prevenire le regressioni

Down e breaking change presso i nostri clienti

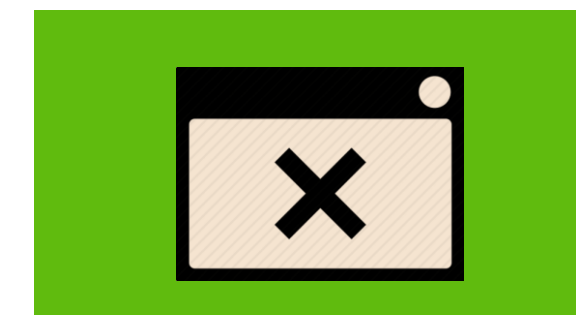
Implementazioni in retro compatibilità

Ridurre i bug e migliorare la qualità

Coprire «in tempo» di test il nostro codice

..con produttività

Tool e framework «integrati» ed integrabili



Non parleremo di..

TDD (Test Driven Development)

Stesura del test prima di quella del codice

Integration test

Test di più componenti in gruppo (integrazione tra i moduli)

Test di UI

Eventi, simulazioni di usabilità su GUI

Processi di automazione

In realtà verrà accennato 😊

Perché unit test

Verifica

Riduzione del numero di bug

Individuazione degli errori

Attinenza alle specifiche richieste

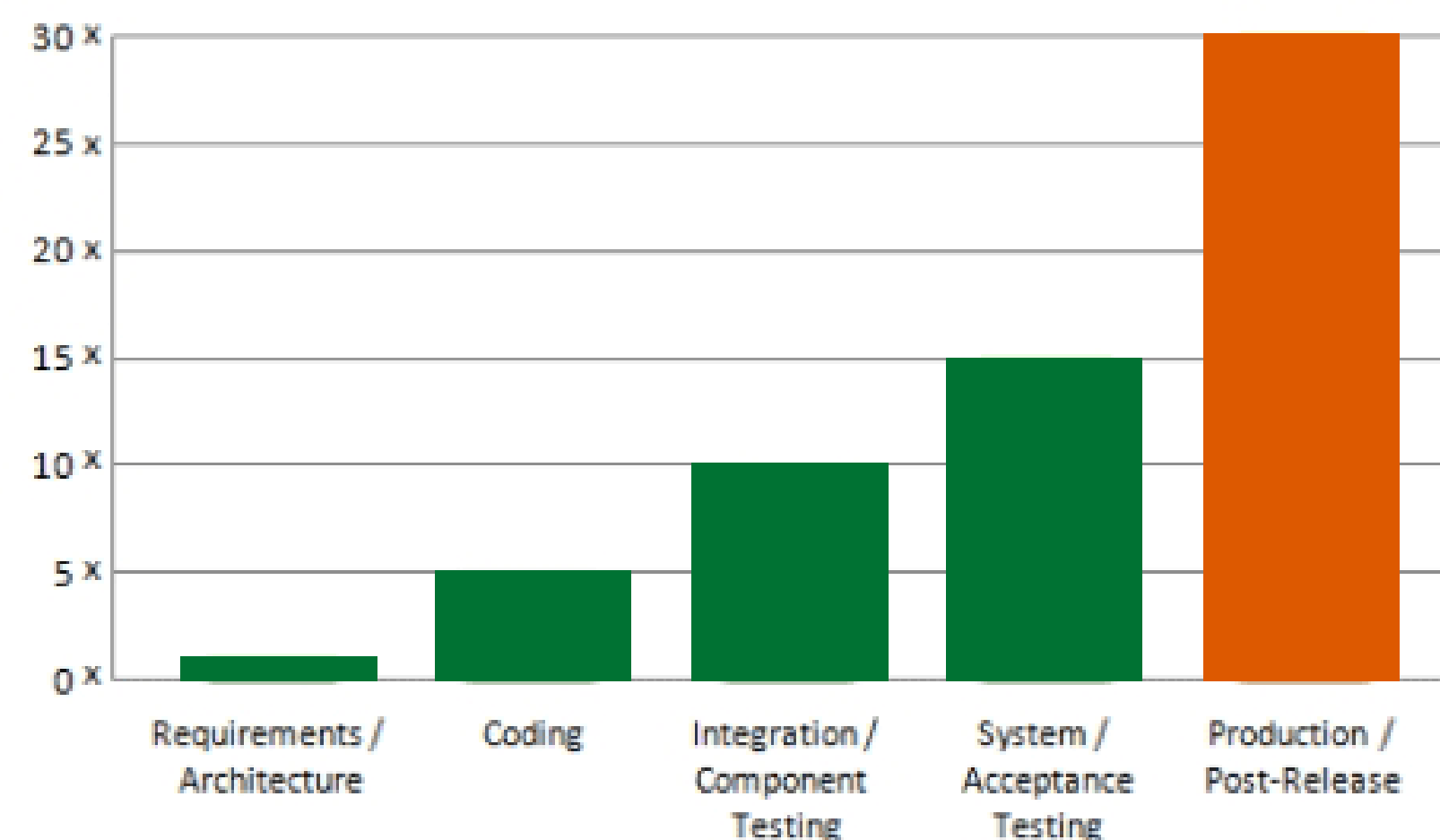
Supporto allo sviluppo evolutivo

Continuous improvement

Riduzione dei costi di sviluppo

Altrimenti impiegato in fix

Relative cost to fix, based on time of detection



Source: National Institute of Standards and Technology

Ma non è solo questo

Altri vantaggi:

Sono automatizzabili (non time consuming) e veloci

Portano «*fiducia*» nel codice prodotto

Conseguentemente, si ha più «*coraggio*» nel fare refactor

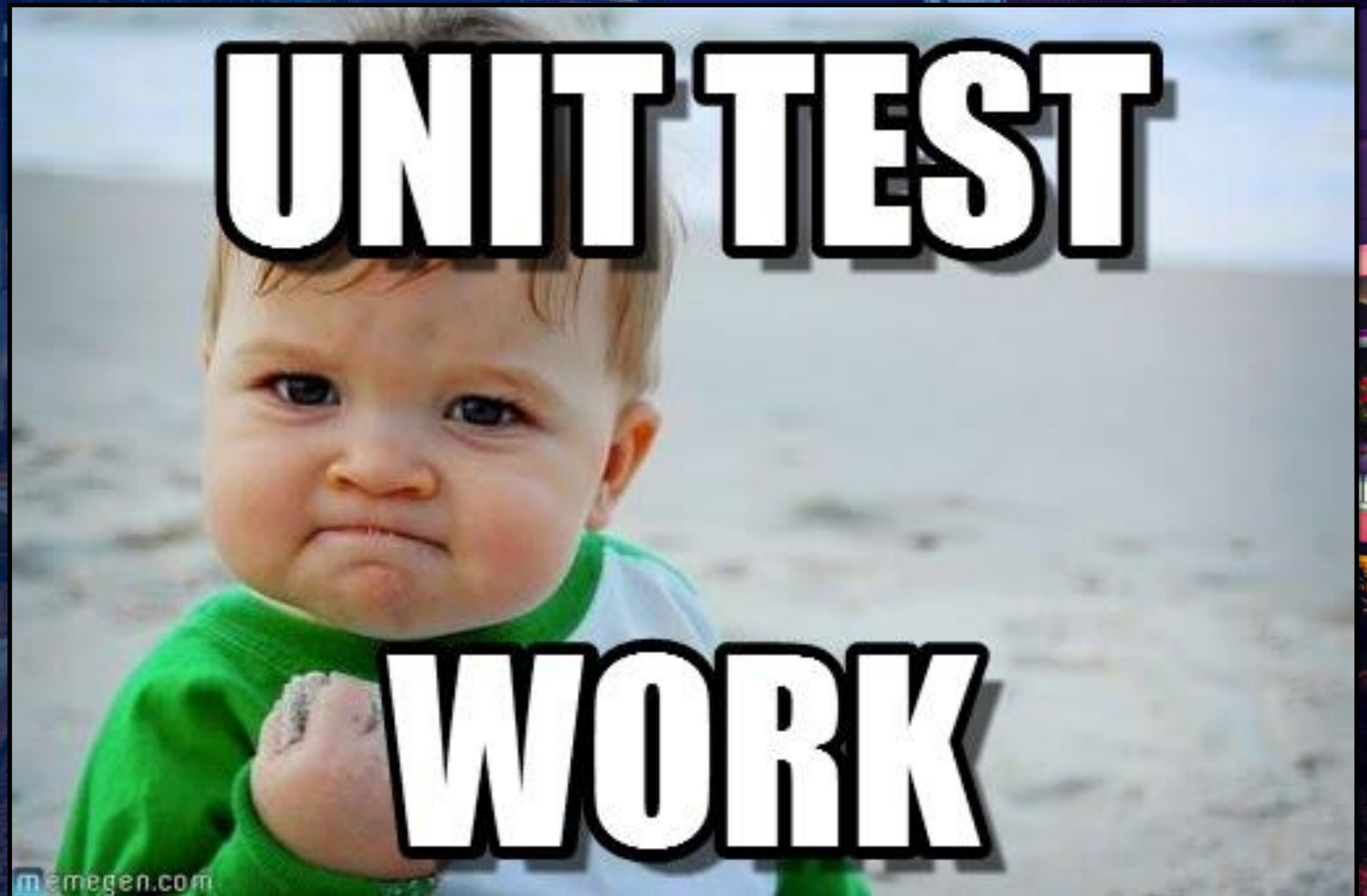
Portano un «*ritmo*» di lavoro, riducendo il tempo del test manuale

Consentono «*stime più precise*» per i cambiamenti

Rendono chiaro il codice

Forniscono una buona documentazione dei metodi

«Fix bugs as soon as you find them»
Non rimandare a domani quello che potresti fare oggi!



www.futuredecoded.it

 #FutureDecoded

Test manuale su database

Esecuzione di codice in ambienti di produzione o estratti di dati

T-SQL debug, check su valori variable

PRINT, PRINT, SELECT...

Non ripetibile, soggetto ad errori umani

Alcuni test non sono più validi appena cambia la situazione

Alcuni test sono fatti su vincoli che bloccano l'operatività del test stesso

Cosa dovremmo testare?

Calcoli in procedure e funzioni

Constraint (schema)

Casi limite di DML e dati

Comportamenti attesi dalle regole sui dati e dai DLM

Error Handling

Security

Standard (SQLCop)

Una soluzione possibile

tSQLt (free)

Semplice da installare

Struttura comune (Assemble, Act, Assert)

Framework in t-sql

...con SQL Server Management Studio

Via t-sql

Con *Red Gate SQL Test*

Integrato anche con il framework SQLCop

Struttura di un test tSQLt

Built-in

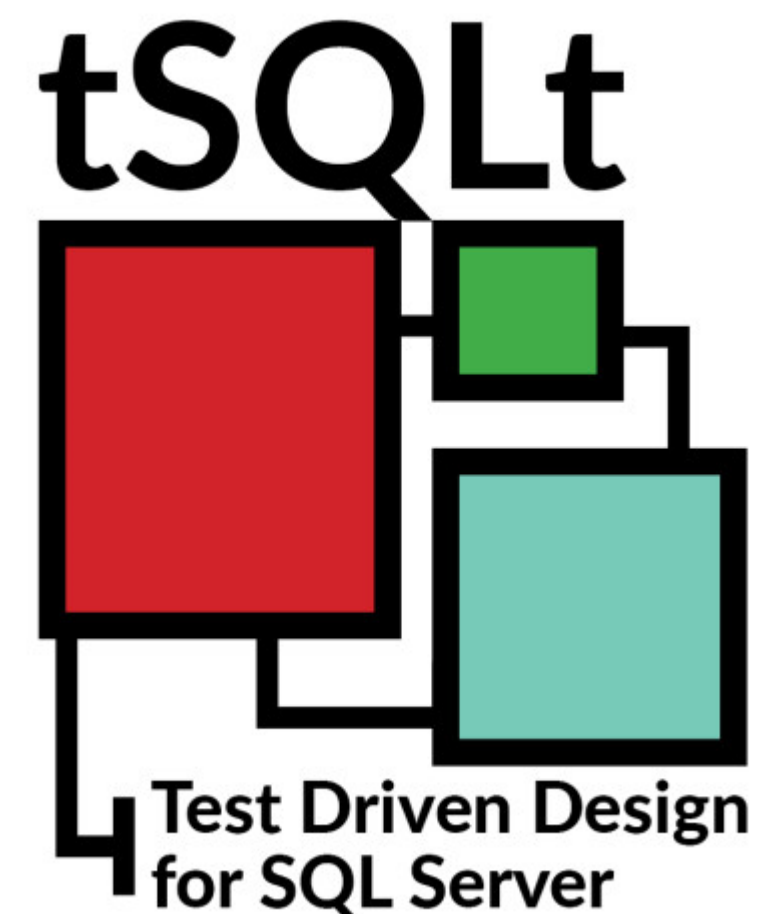
tsqlt schema

Classes

User defined schema

Convenzioni

Naming: test*





DEMO Install tsqit e creazione del primo test

www.futuredecoded.it

 #FutureDecoded

tSQLt test – Pipeline



Features

Assertions

Consentono la comparazione di valori risultanti dalla fase di «ACT»

Isolating dependencies

Consente di isolare le aree di codice che stanno per essere testate

Expectations

Gestione delle eccezioni



DEMO Assemble + Act + Assert

www.futuredecoded.it

 #FutureDecoded

Conseguenze della non prevenzione

I bug non risolti camuffano altri potenziali bug
ed indicano che "la qualità è solo un'opzione"
e sono anche un costo aggiuntivo nel futuro

Discutere dei bug è una perdita di tempo (distrazione del team)

I bug non risolti viziano

le metriche del team e del progetto

le stime preventive e i ritmi di release

Worst practices

Testare procedure troppo complesse

Prima fare refactor, poi testare le unità

Testare più di un comportamento alla volta

Non applicare logiche di business

Testare senza isolare gli oggetti

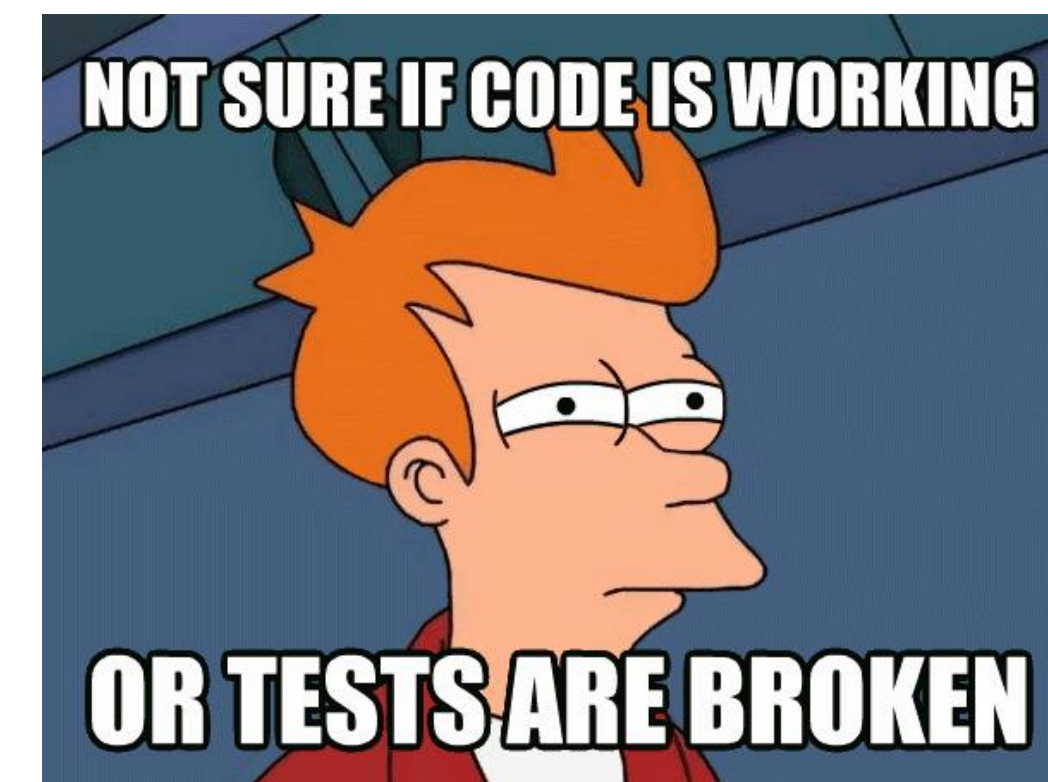
Vizi di integrazione (vincoli, errori fuori dal test)

Fare in modo che il test funzioni ignorando la feature

Quella mania del verde 😊✓✓✓

Test su procedure irrilevanti

Sono solo un costo



Lezioni derivanti

Perciò, coprendo con unit test

Il codice è più leggibile (anche autodocumentato)

Il codice segue regole, e quindi è comprensibile

Il codice diventa quindi familiare, ed è più semplice intervenire

Copertura full vs Copertura parziale

Qualità, l'avevo già detto? 😊

Conclusioni

Nessuna scusa per non testare

Esattamente come per ogni metodo lato codice

I tool esistono

Ed esistono anche i generatori di dati

I processi di test aumentano la qualità

I requisiti di business sono più rispettati e «coperti»

Domande?

Materiale su

<http://www.communitydays.it/>

www.futuredecoded.it



#FutureDecoded



Risorse

<http://www.sqlservercentral.com/articles/Unit+Testing/123900/>

<https://msdn.microsoft.com/it-it/library/mt169842.aspx>

<https://www.youtube.com/watch?v=7tIHg3P0ea0>

<http://tsqlt.org/user-guide/tsqlt-tutorial/>

<http://datacentricity.net/tsqlt/>

<https://www.simple-talk.com/sql/t-sql-programming/getting-started-testing-databases-with-tsqlt/>

<http://www.red-gate.com/products/sql-development/sql-test/>

<https://www.youtube.com/watch?v=l68YO8NyoDc>

<http://blog.apexsql.com/whats-new-in-apexsql-unit-test-2016/>

GRAZIE!
Ci rivediamo fra poco a
Parma!

www.futuredecoded.it

 #FutureDecoded



FUTURE DECODED

6-7 OTT '16 / MILANO

IN PARTNERSHIP WITH:



CommunityDays.it

www.futuredecoded.it



#FutureDecoded