



# Fake the Unfakeable

Isolating Code Under Test with Microsoft Fakes<sup>©</sup>

# Wer bin ich

- Computer-Geek seit 3 Dekaden
- 15 Jahre Professional – 6 davon selbständig
- Horror-Film Fan
- Monk (ig)
- Straight Edge
- Vegetarier



# [Setup]

- Theorie
  - *Was / Warum / Wie*
- Praxis
  1. *Configuration*
  2. *Casuals (DateTime, Events, Statics, ctor ...)*
  3. *Datenbank (Entity Framework)*
  4. *HttpContext*

# Was

... ist Microsoft Fakes<sup>©</sup>?

„Code Isolation Framework zur Aufbereitung des Application-Codes für das Unit Testing“

State Verification [\[xunitpatterns\]](#)

# Warum

... wurde Microsoft Fakes<sup>©</sup> entwickelt?

- Entwicklung == Evolution ... auch bei Microsoft 😊
- Optimales Laufzeitverhalten <> Testbarkeit
- Code Qualität && Komponenten/Module

# Wie

... arbeitet Microsoft Fakes<sup>©</sup>?

- Begrifflichkeiten
  - *Stubs*
  - *Shims*
- Integration
  - *Konfiguration*
  - *Kompilation*
  - *Testing*

# Stubs

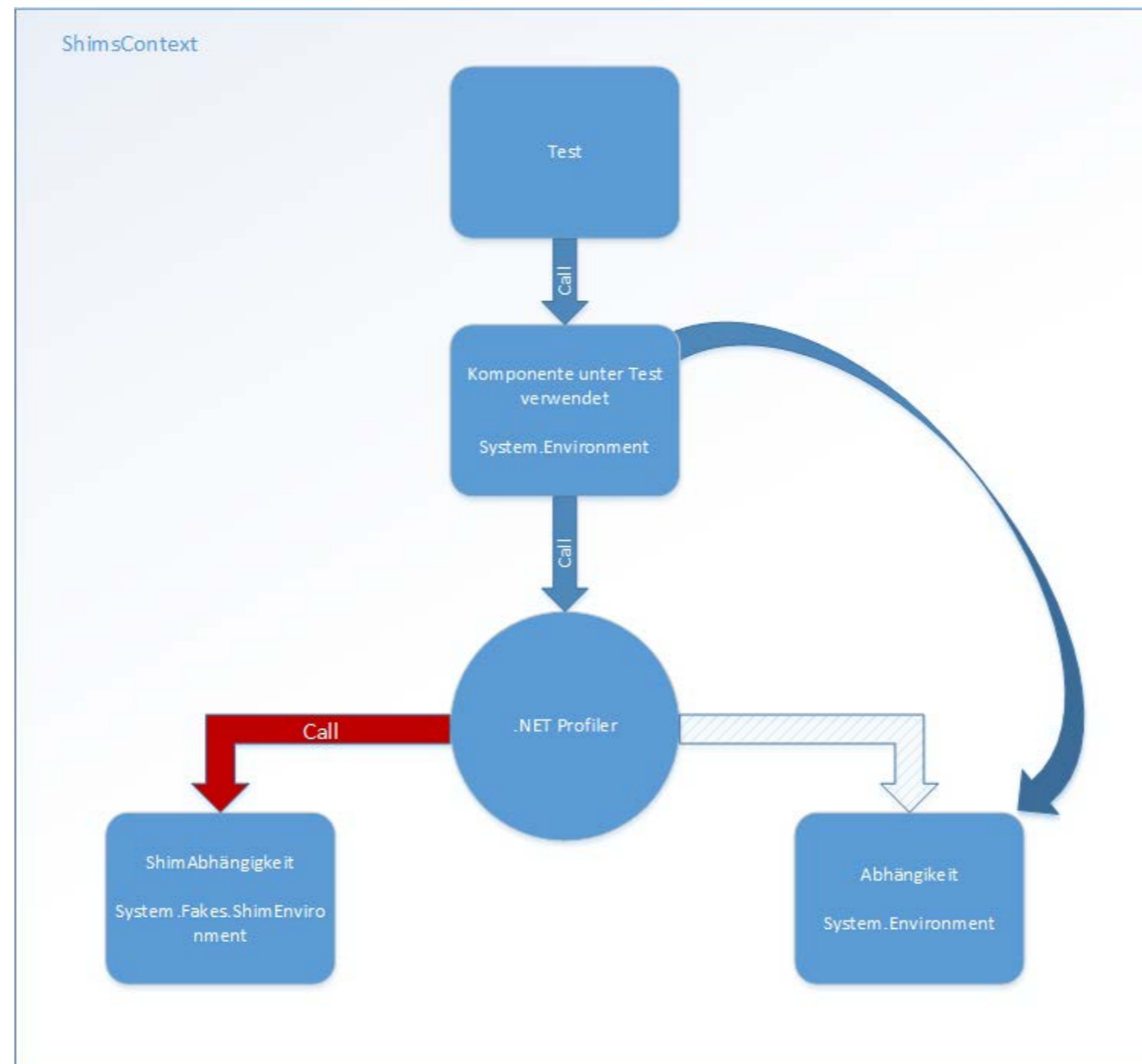
- Substitution funktionaler Bausteine [\[MSDN\]](#)
  - *Substitution kann „injected“ werden*
  - *„IOC“-like*
  - *Anwendung auf ein Interface*
  - *Pointer Signaturen werden nicht unterstützt!*

# Shims |

- Runtime Interception [\[MSDN\]](#)
  - *Substitution kann nicht „injected“ werden*
  - *.NET Profiler leitet Aufrufe um („Monkey Patching“)*



# Shims II



# Konfiguration

- .fakes File in der Solution [\[MSDN\]](#)
  - *Konfiguration: Kompilation, Logging ...*
  - *Filtern: assemblies, namespaces, types ...*
  - *case-sensitive: !, StartsWith: \*, Aufzählung: ;*
- Microsoft.QualityTools.Testing.Fakes.targets
  - *nicht offiziell dokumentiert* [\[Blog\]](#)

# Kompilation

- Assembly: `<assembly>. <version>.Fakes`
- Namespace: `<namespace>.Fakes`
- Namenskonventionen: [\[MSDN\]](#)
  - *Shim / Stub Prefix*
  - *Methoden-Postfix (Get/Set/Constructor...)*
  - *Parameter-Postfix (T/TOut/TRef/TArray...)*

# Testing

- Stubs: Ausführung ohne Einschränkung
- Shims: benötigen immer einen ShimsContext
  - *Call-Interception und Umlenkung aller Aufrufe innerhalb des ShimsContext*
  - *.ExecuteWithoutShims() möglich*

# Praxisbeispiele

... na endlich 😊

# [TearDown]

- Fragen?
  - *Jetzt an mich oder später an:*  
[kimpenhaus@devil-engineering.de](mailto:kimpenhaus@devil-engineering.de)
- Folien: <http://www.5Minds.de/events.html>
- Code: [https://github.com/5minds/fe\\_fakes](https://github.com/5minds/fe_fakes)

Danke für Eure Aufmerksamkeit!