


Erste Erfahrungen mit Swift Testing

Überblick


Aufgeräumte Swift-API

```
struct GreeterTest {  
    @Test func greetingMessage() {  
        ...  
        #expect(greeter.message == "Hello World")  
    }  
}
```

Integration in Xcode

▼  GreeterTest



 greetingMessage()



```
❌ struct GreeterTest {  
❌     @Test func greetingMessage() {  
49         let greeter = Greeter()  
50         #expect(greeter.message == "Hello World")
```

⌕ Results

greeter.message : "Hello Wrold"

String

_ : "Hello World"

String

```
51     }
```

Assertions

- `#expect(pi > 3)`
- `#expect(x != nil)`
- `#expect(throws: FooError) { try foo() }`
- `let unwrapped = try #require(optional)`
- `Issue.record("Reference is not a tree!")`

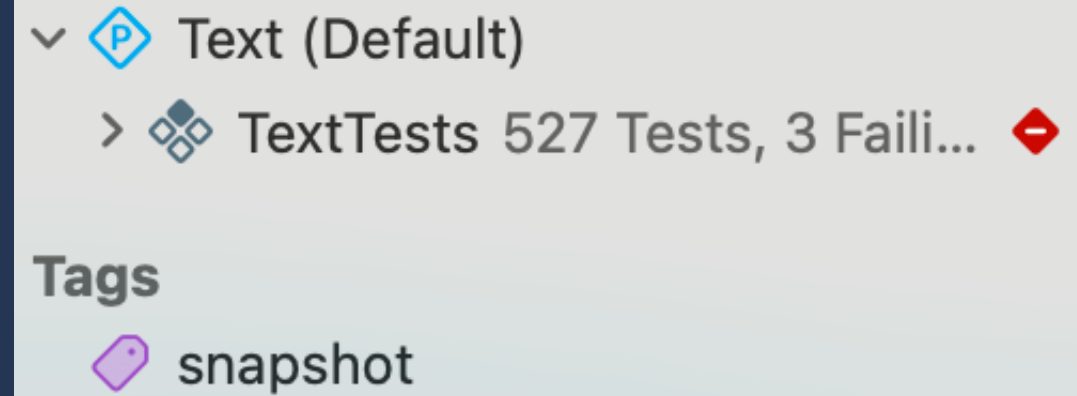
Bedingte Tests

```
@Test(enabled(if: Global.cloudKitAvailable))  
func sync() async throws {  
    ...  
}
```


Tagging

```
@Test(.tags(.snapshot))
func whatever() {
    ...
}

// Definition
public extension Tags {
    @Tag static var snapshot: Self
}
```



Argumente

```
@Test(arguments: [7.0, 11.0, 18.0])  
func fontSizes(size: Double) {  
    render(fontSize: size)  
    ...  
}
```

✓	testFontSizes(size:)	✓
n	7.0	✓
n	11.0	✓
n	18.0	✓

Suites

```
@Suite(.tags(snapshot))
struct RendererTests {
    @Test func images() { ... }
    @Test func table() { ... }
}
```

Setup / Teardown

```
class MyTest {  
    init() { ... }  
    deinit { ... }  
}
```

Parallel by Default

Async

```
@Test func notifies() async {  
    ...  
    await confirmation { confirmation in  
        // Event-Handler  
        cellbroadcast.onReceive = { confirmation() }  
  
        // Auslöser  
        cellbroadcast.notify("Warntag! Keine Gefahr!")  
    }  
}
```

Erfahrungen ...

Struct Teardown?

```
struct MyTest {  
    init() { ... }  
}
```

- Über Reference-Type-Property
- Async Schwierig
- Zeitpunkt generell unsicher (Memory Management...)

Async Teardown?

```
class MyTest {  
    deinit { ... }  
}
```

Tags können vergessen werden...

```
@Test(.tags(.snapshot))  
func rendering() {  
    validateSnapshot(...)  
}
```

Tag-Assertions

```
@Test(.tags(.snapshot))  
func rendering() {  
    validateSnapshot(...)  
}
```

// Assertions in Test-Infrastruktur

```
func validateSnapshot(...) {  
    guard Set(Test.current.tags).contains(.snapshot) else {  
        throw ...  
    }  
}
```

Macros, Macros, Macros 🍌

- `try #expect(doSomething())` ❌
- `#expect(try doSomething)` ✅

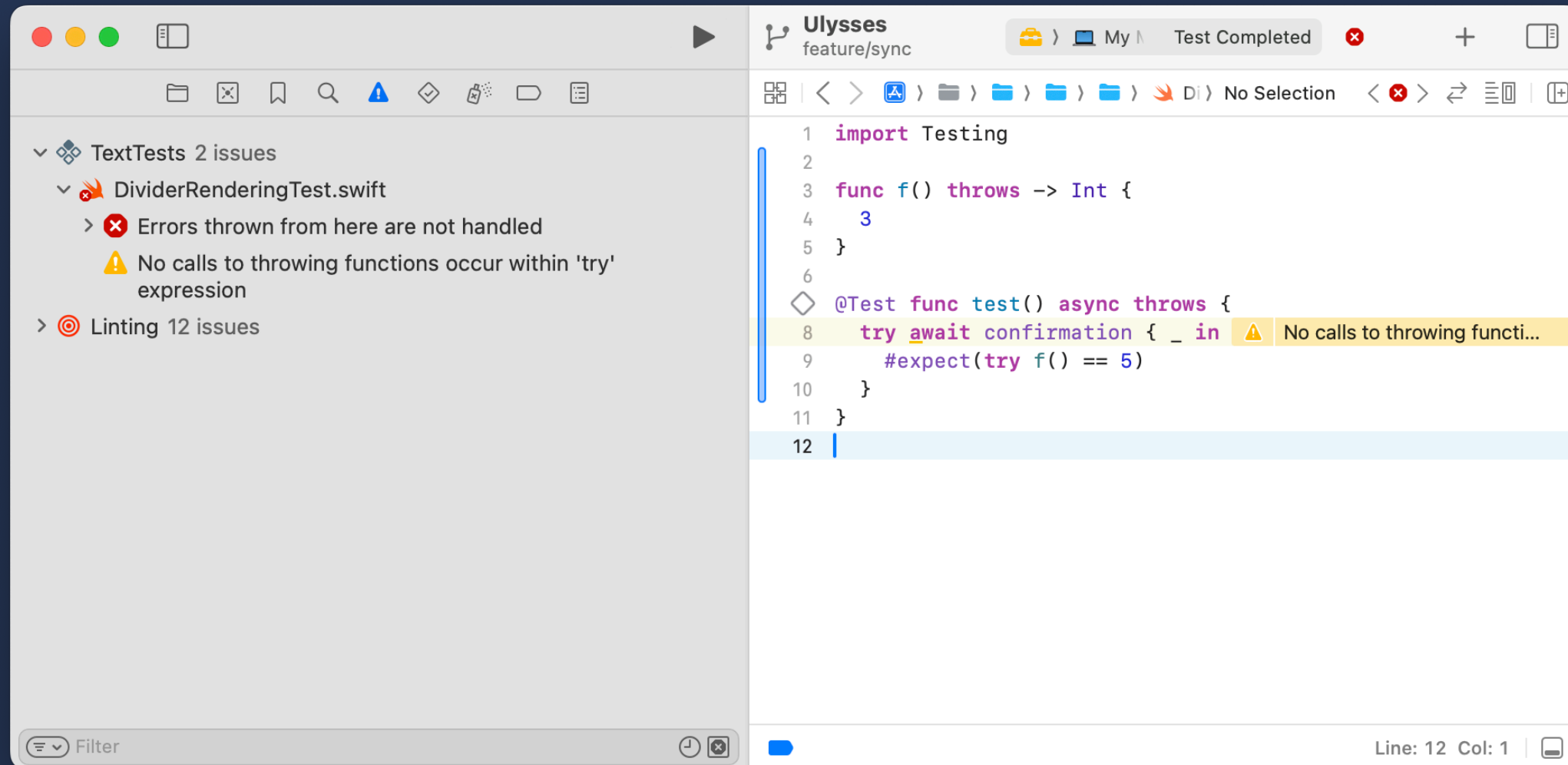
Rethrows & Expect 🍌

```
try await confirmation { onConfirm in  
    #expect(try foo())  
}
```

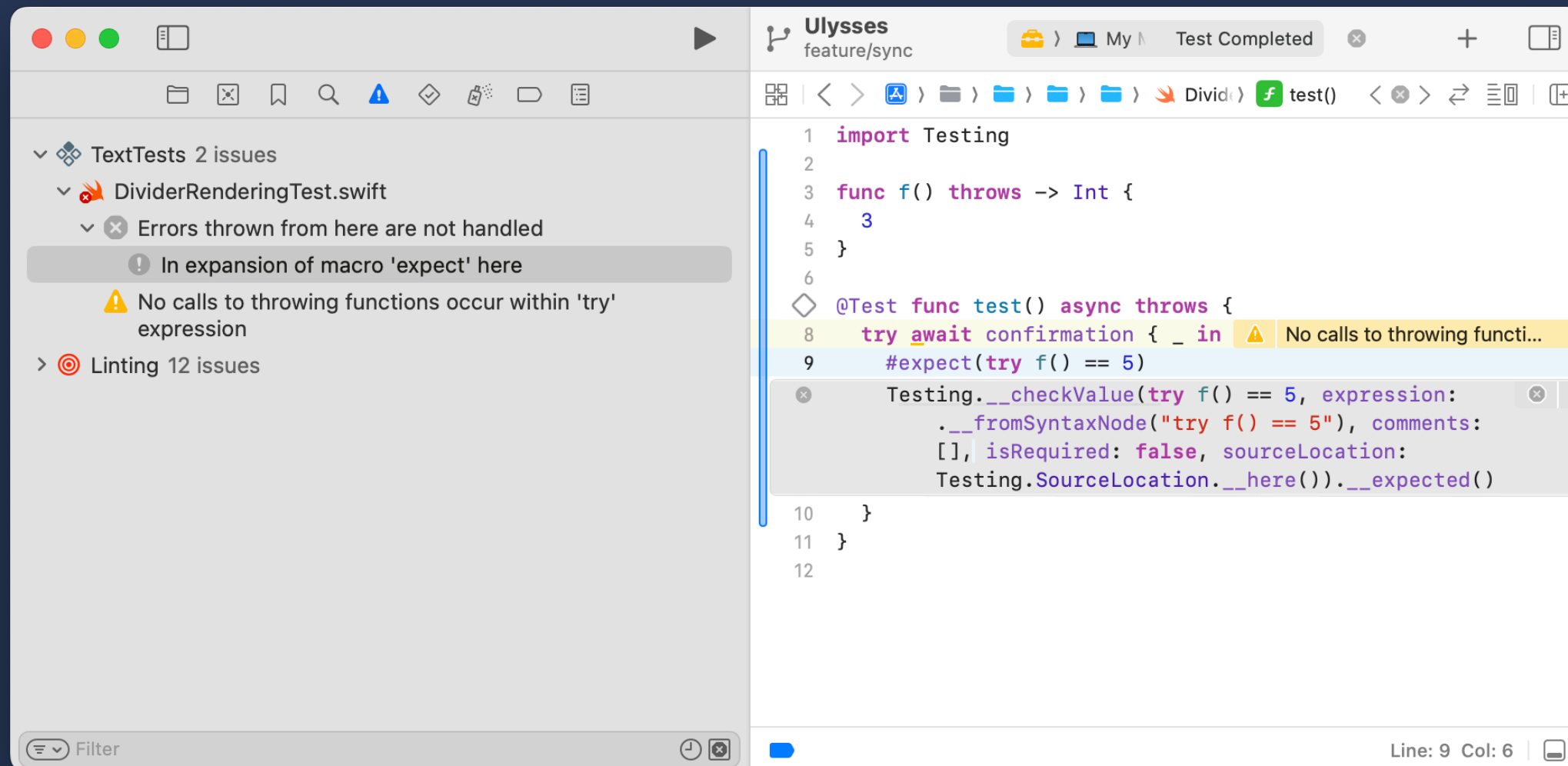
Rethrows & Expect

```
try await confirmation { onConfirm in
    let x = try foo()
    #expect(x)
}
```

Compiler errors 🍌



Compiler errors 🍌



Stack traces 🍌

The image displays a Swift stack trace from Xcode on the left and the Ulysses app on the right, illustrating a failing test case.

Stack Trace (Xcode):

- Thread 7 Queue: com.apple.root.user-initiated-qos.cooperative (concurrent)
- 0 GreeterTest.greetingMessage() [inlined]
- 1 GreeterTest.greetingMessage()
- 2 closure #2 in static GreeterTest.\$s9TextTests11GreeterTestV15greetingMessage0D0fMp_21funcgreetingMessage__fMu0_@Sendable ()
- 3 (3) await resume partial function for Testing.__ifMainActorIsolationEnforced<τ_0_0 where τ_0_0: Swift.Sendable>(_: @Swift.MainActor @Sendable () async throws -> τ_0_0, else: @Sendable () async throws -> τ_0_0) async throws -> τ_0_0
- 4 static GreeterTest.\$s9TextTests11GreeterTestV15greetingMessage0D0fMp_21funcgreetingMessage__fMu0_@Sendable ()
- 5 implicit closure #1 in static GreeterTest.\$s9TextTests11GreeterTestV15greetingMessage0D0fMp_51_...Test_container__function__funcgreetingMessage__fMu0_...tests.getter
- 6 (4) await resume partial function for closure #1 () async throws -> () in closure #1 () async -> () in Testing.Runner._runTestCase(_: Testing.Test.Case, within: Testing.Runner.Plan.Step) async throws -> ()
- 7 (1) await resume partial function for partial apply forwarder for closure #1 @Sendable () async throws -> () in closure #1 @Sendable ((τ_1_0, τ_1_1)) -> Testing.Test.Case in Testing.Test.Case.Generator.init<τ_0_0, τ_0_1 where τ_1_0: Swift.Sendable, τ_1_1: Swift.Sendable, ...
- 8 (2) await resume partial function for function signature specialization <Arg[3] = Dead> of static Testing.Issue.withErrorRecording(at: Testing.SourceLocation, configuration: Swift.Optional<Testing.Configuration>, _: () async throws -> ()) async -> Swift.Optional<Swift.Err...
- 9 (2) await resume partial function for closure #1 () async -> () in Testing.Runner._runTestCase(_: Testing.Test.Case, within: Testing.Runner.Plan.Step) async throws -> ()
- 10 (1) await resume partial function for partial apply forwarder for closure #1 @Sendable () async throws -> () in closure #1 @Sendable ((τ_1_0, τ_1_1)) -> Testing.Test.Case in Testing.Test.Case.Generator.init<τ_0_0, τ_0_1 where τ_1_0: Swift.Sendable, τ_1_1: Swift.Sendable, ...
- 11 (2) await resume partial function for Swift.Task.Local.withValueImpl<τ_0_0>(_: __owned τ_0_0, operation: () async throws -> τ_1_0, isolation: isolated Swift.Optional<Swift.Actor>, file: Swift.String, line: Swift.UInt) async throws -> τ_1_0
- 12 (2) await resume partial function for Swift.Task.Local.withValue<τ_0_0>(_: τ_0_0, operation: () async throws -> τ_1_0, isolation: isolated Swift.Optional<Swift.Actor>, file: Swift.String, line: Swift.UInt) async throws -> τ_1_0
- 13 (2) await resume partial function for generic specialization <()> of static Testing.Test.Case.withCurrent<τ_0_0>(_: Testing.Test.Case, perform: () async throws -> τ_0_0) async throws -> τ_0_0
- 14 (13) await resume partial function for generic specialization <Swift.AnySequence<Testing.Test.Case>> of Testing.Runner._runTestCases<τ_0_0 where τ_0_0: Swift.Sequence, τ_0_0.Element == Testing.Test.Case>(_: τ_0_0, within: Testing.Runner.Plan.Step) async throws...
- 15 (2) await resume partial function for closure #1 @Sendable () async throws -> () in closure #1 () async throws -> () in Testing.Runner._runStep(atRootOf: Testing.Graph<Swift.String, Swift.Optional<Testing.Runner.Plan.Step>>, depth: Swift.Int, ...
- 16 (2) await resume partial function for closure #1 () async throws -> () in closure #1 () async -> () in Testing.Runner._runStep(atRootOf: Testing.Graph<Swift.String, Swift.Optional<Testing.Runner.Plan.Step>>, depth: Swift.Int, lastAncestorStep: Swift.Optional<Testing.Run...
- 17 (1) await resume partial function for partial apply forwarder for closure #1 @Sendable () async throws -> () in closure #1 @Sendable ((τ_1_0, τ_1_1)) -> Testing.Test.Case in Testing.Test.Case.Generator.init<τ_0_0, τ_0_1 where τ_1_0: Swift.Sendable, τ_1_1: Swift.Sendable, ...
- 18 (2) await resume partial function for function signature specialization <Arg[3] = Dead> of static Testing.Issue.withErrorRecording(at: Testing.SourceLocation, configuration: Swift.Optional<Testing.Configuration>, _: () async throws -> ()) async -> Swift.Optional<Swift.Err...
- 19 (2) await resume partial function for closure #1 () async -> () in Testing.Runner._runStep(atRootOf: Testing.Graph<Swift.String, Swift.Optional<Testing.Runner.Plan.Step>>, depth: Swift.Int, lastAncestorStep: Swift.Optional<Testing.Runner.Plan.Step>) async throws -> ()
- 20 (1) await resume partial function for partial apply forwarder for closure #1 @Sendable () async throws -> () in closure #1 @Sendable ((τ_1_0, τ_1_1)) -> Testing.Test.Case in Testing.Test.Case.Generator.init<τ_0_0, τ_0_1 where τ_1_0: Swift.Sendable, τ_1_1: Swift.Sendable, ...
- 21 (2) await resume partial function for Swift.Task.Local.withValueImpl<τ_0_0>(_: __owned τ_0_0, operation: () async throws -> τ_1_0, isolation: isolated Swift.Optional<Swift.Actor>, file: Swift.String, line: Swift.UInt) async throws -> τ_1_0
- 22 (2) await resume partial function for Swift.Task.Local.withValue<τ_0_0>(_: τ_0_0, operation: () async throws -> τ_1_0, isolation: isolated Swift.Optional<Swift.Actor>, file: Swift.String, line: Swift.UInt) async throws -> τ_1_0
- 23 (2) await resume partial function for generic specialization <()> of static Testing.Test.withCurrent<τ_0_0>(_: Testing.Test, perform: () async throws -> τ_0_0) async throws -> τ_0_0
- 24 (2) await resume partial function for Testing.Runner._runStep(atRootOf: Testing.Graph<Swift.String, Swift.Optional<Testing.Runner.Plan.Step>>, depth: Swift.Int, lastAncestorStep: Swift.Optional<Testing.Runner.Plan.Step>) async throws -> ()
- 25 (2) await resume partial function for closure #2 @Sendable (Swift.String, Testing.Graph<Swift.String, Swift.Optional<Testing.Runner.Plan.Step>>) async throws -> () in Testing.Runner._runChildren(of: Testing.Graph<Swift.String, Swift.Optional<Testing.Runner.Plan.Step...)
- 26 (1) await resume partial function for partial apply forwarder for closure #1 @Sendable () async throws -> () in closure #1 @Sendable ((τ_1_0, τ_1_1)) -> Testing.Test.Case in Testing.Test.Case.Generator.init<τ_0_0, τ_0_1 where τ_1_0: Swift.Sendable, τ_1_1: Swift.Sendable, ...
- 27 (13) await resume partial function for function signature specialization <Arg[2] = [Closure Propagated : reabstraction thunk helper from @escaping @callee_guaranteed @Sendable @async (@guaranteed Swift.String, @in_guaranteed Testing.Graph<Swift.String, Swift.O...
- 28 (2) await resume partial function for Testing.Runner._runChildren(of: Testing.Graph<Swift.String, Swift.Optional<Testing.Runner.Plan.Step>>, depth: Swift.Int, lastAncestorStep: Swift.Optional<Testing.Runner.Plan.Step>) async throws -> ()
- 29 (4) await resume partial function for Testing.Runner._runStep(atRootOf: Testing.Graph<Swift.String, Swift.Optional<Testing.Runner.Plan.Step>>, depth: Swift.Int, lastAncestorStep: Swift.Optional<Testing.Runner.Plan.Step>) async throws -> ()
- 30 (2) await resume partial function for closure #2 @Sendable (Swift.String, Testing.Graph<Swift.String, Swift.Optional<Testing.Runner.Plan.Step>>) async throws -> () in Testing.Runner._runChildren(of: Testing.Graph<Swift.String, Swift.Optional<Testing.Runner.Plan.Step...)
- 31 (1) await resume partial function for partial apply forwarder for closure #1 @Sendable () async throws -> () in closure #1 @Sendable ((τ_1_0, τ_1_1)) -> Testing.Test.Case in Testing.Test.Case.Generator.init<τ_0_0, τ_0_1 where τ_1_0: Swift.Sendable, τ_1_1: Swift.Sendable, ...
- 32 (13) await resume partial function for function signature specialization <Arg[2] = [Closure Propagated : reabstraction thunk helper from @escaping @callee_guaranteed @Sendable @async (@guaranteed Swift.String, @in_guaranteed Testing.Graph<Swift.String, Swift.O...
- 33 (2) await resume partial function for Testing.Runner._runChildren(of: Testing.Graph<Swift.String, Swift.Optional<Testing.Runner.Plan.Step>>, depth: Swift.Int, lastAncestorStep: Swift.Optional<Testing.Runner.Plan.Step>) async throws -> ()
- 34 (4) await resume partial function for closure #1 @Sendable () async throws -> () in closure #1 () async -> () in Testing.Runner._runStep(atRootOf: Testing.Graph<Swift.String, Swift.Optional<Testing.Runner.Plan.Step>>, depth: Swift.Int, ...
- 35 (2) await resume partial function for closure #1 () async throws -> () in closure #1 () async -> () in Testing.Runner._runStep(atRootOf: Testing.Graph<Swift.String, Swift.Optional<Testing.Runner.Plan.Step>>, depth: Swift.Int, lastAncestorStep: Swift.Optional<Testing.Run...
- 36 (1) await resume partial function for partial apply forwarder for closure #1 @Sendable () async throws -> () in closure #1 @Sendable ((τ_1_0, τ_1_1)) -> Testing.Test.Case in Testing.Test.Case.Generator.init<τ_0_0, τ_0_1 where τ_1_0: Swift.Sendable, τ_1_1: Swift.Sendable, ...
- 37 (2) await resume partial function for function signature specialization <Arg[3] = Dead> of static Testing.Issue.withErrorRecording(at: Testing.SourceLocation, configuration: Swift.Optional<Testing.Configuration>, _: () async throws -> ()) async -> Swift.Optional<Swift.Err...
- 38 (2) await resume partial function for closure #1 () async -> () in Testing.Runner._runStep(atRootOf: Testing.Graph<Swift.String, Swift.Optional<Testing.Runner.Plan.Step>>, depth: Swift.Int, lastAncestorStep: Swift.Optional<Testing.Runner.Plan.Step>) async throws -> ()
- 39 (1) await resume partial function for partial apply forwarder for closure #1 @Sendable () async throws -> () in closure #1 @Sendable ((τ_1_0, τ_1_1)) -> Testing.Test.Case in Testing.Test.Case.Generator.init<τ_0_0, τ_0_1 where τ_1_0: Swift.Sendable, τ_1_1: Swift.Sendable, ...
- 40 (2) await resume partial function for Swift.Task.Local.withValueImpl<τ_0_0>(_: __owned τ_0_0, operation: () async throws -> τ_1_0, isolation: isolated Swift.Optional<Swift.Actor>, file: Swift.String, line: Swift.UInt) async throws -> τ_1_0
- 41 (2) await resume partial function for Swift.Task.Local.withValue<τ_0_0>(_: τ_0_0, operation: () async throws -> τ_1_0, isolation: isolated Swift.Optional<Swift.Actor>, file: Swift.String, line: Swift.UInt) async throws -> τ_1_0
- 42 (2) await resume partial function for generic specialization <()> of static Testing.Test.withCurrent<τ_0_0>(_: Testing.Test, perform: () async throws -> τ_0_0) async throws -> τ_0_0
- 43 (2) await resume partial function for Testing.Runner._runStep(atRootOf: Testing.Graph<Swift.String, Swift.Optional<Testing.Runner.Plan.Step>>, depth: Swift.Int, lastAncestorStep: Swift.Optional<Testing.Runner.Plan.Step>) async throws -> ()
- 44 (2) await resume partial function for closure #2 @Sendable (Swift.String, Testing.Graph<Swift.String, Swift.Optional<Testing.Runner.Plan.Step>>) async throws -> () in Testing.Runner._runChildren(of: Testing.Graph<Swift.String, Swift.Optional<Testing.Runner.Plan.Step...)
- 45 (1) await resume partial function for partial apply forwarder for closure #1 @Sendable () async throws -> () in closure #1 @Sendable ((τ_1_0, τ_1_1)) -> Testing.Test.Case in Testing.Test.Case.Generator.init<τ_0_0, τ_0_1 where τ_1_0: Swift.Sendable, τ_1_1: Swift.Sendable, ...
- 46 (13) await resume partial function for function signature specialization <Arg[2] = [Closure Propagated : reabstraction thunk helper from @escaping @callee_guaranteed @Sendable @async (@guaranteed Swift.String, @in_guaranteed Testing.Graph<Swift.String, Swift...
- 47 (2) await resume partial function for Testing.Runner._runChildren(of: Testing.Graph<Swift.String, Swift.Optional<Testing.Runner.Plan.Step>>, depth: Swift.Int, lastAncestorStep: Swift.Optional<Testing.Runner.Plan.Step>) async throws -> ()
- 48 (4) await resume partial function for Testing.Runner._runStep(atRootOf: Testing.Graph<Swift.String, Swift.Optional<Testing.Runner.Plan.Step>>, depth: Swift.Int, lastAncestorStep: Swift.Optional<Testing.Runner.Plan.Step>) async throws -> ()
- 49 (2) await resume partial function for closure #2 @Sendable (Swift.String, Testing.Graph<Swift.String, Swift.Optional<Testing.Runner.Plan.Step>>) async throws -> () in Testing.Runner._runChildren(of: Testing.Graph<Swift.String, Swift.Optional<Testing.Runner.Plan.Step...)
- 50 (1) await resume partial function for partial apply forwarder for closure #1 @Sendable () async throws -> () in closure #1 @Sendable ((τ_1_0, τ_1_1)) -> Testing.Test.Case in Testing.Test.Case.Generator.init<τ_0_0, τ_0_1 where τ_1_0: Swift.Sendable, τ_1_1: Swift.Sendable, ...
- 51 (13) await resume partial function for function signature specialization <Arg[2] = [Closure Propagated : reabstraction thunk helper from @escaping @callee_guaranteed @Sendable @async (@guaranteed Swift.String, @in_guaranteed Testing.Graph<Swift.String, Swift.O...
- 52 (2) await resume partial function for Testing.Runner._runChildren(of: Testing.Graph<Swift.String, Swift.Optional<Testing.Runner.Plan.Step>>, depth: Swift.Int, lastAncestorStep: Swift.Optional<Testing.Runner.Plan.Step>) async throws -> ()
- 53 (4) await resume partial function for Testing.Runner._runStep(atRootOf: Testing.Graph<Swift.String, Swift.Optional<Testing.Runner.Plan.Step>>, depth: Swift.Int, lastAncestorStep: Swift.Optional<Testing.Runner.Plan.Step>) async throws -> ()
- 54 (2) await resume partial function for closure #1 @Sendable () async -> () in closure #1 (inout Swift.TaskGroup<()>) async -> () in closure #2 () async -> () in static Testing.Runner._run(Testing.Runner) async -> ()
- 55 (1) await resume partial function for partial apply forwarder for closure #1 @Sendable () async throws -> () in closure #1 @Sendable ((τ_1_0, τ_1_1)) -> Testing.Test.Case in Testing.Test.Case.Generator.init<τ_0_0, τ_0_1 where τ_1_0: Swift.Sendable, τ_1_1: Swift.Sendable, ...
- 56 (1) await resume partial function for generic specialization <()> of reabstraction thunk helper <τ_0_0 where τ_0_0: Swift.Sendable> from @escaping @isolated(any) @callee_guaranteed @Sendable @async () -> (@out τ_0_0) to @escaping @callee_guaranteed @async...
- 57 (1) await resume partial function for partial apply forwarder for closure #1 @Sendable () async throws -> () in closure #1 @Sendable ((τ_1_0, τ_1_1)) -> Testing.Test.Case in Testing.Test.Case.Generator.init<τ_0_0, τ_0_1 where τ_1_0: Swift.Sendable, τ_1_1: Swift.Sendable, ...

Ulysses App:

```
1 import Testing
2
3 struct Greeter {
4   let message = "Hello World"
5 }
6
7 struct GreeterTest {
8   @Test func greetingMessage() {
9     let greeter = Greeter()
10    #expect{
11      greeter.message == "true"
12    }
13  }
14 }
15
```

Test Results:

- Test Suite 'Selected tests' started at 2024-09-12 15:35:44.428.
- Test Suite 'TextTests.xctest' started at 2024-09-12 15:35:44.429.
- Test Suite 'TextTests.xctest' passed at 2024-09-12 15:35:44.429.
- Executed 0 tests, with 0 failures (0 unexpected) in 0.000 (0.000) seconds
- Test Suite 'Selected tests' passed at 2024-09-12 15:35:44.429.
- Executed 0 tests, with 0 failures (0 unexpected) in 0.000 (0.001) seconds
- Test run started.
- Testing Library Version: 92 (arm64e-apple-macos13.0)
- Suite GreeterTest started.
- Test greetingMessage() started.
- Test greetingMessage() recorded an issue at DividerRenderingTest.swift:10:3: Expectation failed: (greeter.message → "Hello World") == "true"

Max

Und ihr so?