



Particularidades de la asincronía

Antes de entrar en materia, nos debemos preguntar cuál es el problema que estamos intentando resolver. En este caso es el de evitar bloquear nuestro hilo principal de ejecución mientras se realizan operaciones de entrada y salida a la base de datos. **La asincronía responde por tanto una necesidad a nivel de rendimiento.**

Futuros de Scala

Gracias a la gestión nativa por parte de Scala de los Futuros, no necesitaremos guardar nuestras peticiones para ser procesadas en background por un proceso secundario. Scala ya nos provee de un contexto de ejecución con múltiples hilos a los que podemos asignar tareas para evitar bloquear el hilo principal.

Con esto conseguimos que, de forma sencilla desde el punto de vista del programador, podamos explotar todas las funcionalidades a nivel de rendimiento que ofrezca nuestro hardware.

Test

Aquí podemos ver uno de los test mostrados en el vídeo. En este test se puede apreciar cómo al ejecutarlo, veremos el texto "Query asked to the async query bus" antes de pintar el texto desde el repositorio.

```
package tv.codely.cqrs_ddd_scala_example.acceptance

import java.util.UUID

import scala.reflect.ClassTag
import scala.concurrent.duration._
import scala.concurrent.ExecutionContext.Implicits.global

import cats.implicits._
import org.joda.time.DateTime
import org.scalatest.concurrent.ScalaFutures
import org.scalatest.Matchers._
import org.scalatest._
import org.scalatest.concurrent.PatienceConfiguration.Timeout
import org.scalatest.time.{Seconds, Span}
import tv.codely.cqrs_ddd_scala_example.bus.domain.QueryBus
import tv.codely.cqrs_ddd_scala_example.user_greet.application.generate.{
  FindUserGreetQuery,
  FindUserGreetQueryHandler,
  UserGreetFinder
}
import tv.codely.cqrs_ddd_scala_example.user_greet.infrastructure.InAsyncDelayedMemory

final class AsyncUserGreetFinderTest extends WordSpec with ScalaFutures with GivenWhen

  "UserGreetGenerator with an AsyncQueryBus" should {
    "not block the execution flow until getting a response from a slow repository" in

      Given("a UserGreetGenerator with a user repository with a notable delay")

        val notableDelay = 10.seconds
        val userRepository = new InAsyncDelayedMemoryUserRepository(notableDelay)
        val userGreetGeneratorWithDelay = new UserGreetFinder(userRepository)
        val generateUserGreetQueryHandler = new FindUserGreetQueryHandler(userGreetGeneratorWithDelay)

        And("an AsyncQueryBus which doesn't block the execution flow until getting a response from a slow repository")

        val queryBus = new QueryBus(
          Map(
            classTag[FindUserGreetQuery] -> generateUserGreetQueryHandler
          )
        )

        When("we ask the GenerateUserGreetQuery to the AsyncQueryBus")

        val query = FindUserGreetQuery(
          UUID.randomUUID(),
          DateTime.now(),
          UUID.fromString("1646fd5c-de2b-435f-b20f-ad1f50924dfe")
        )

        val futureGreeting = queryBus.ask(query)

        println("Query asked to the async query bus")
  }
```

```
Then("it should say hello to someone")

    futureGreeting.futureValue(Timeout(Span(15, Seconds))).greet
}
}
```

Enlaces de interés

- [Repositorio con los ejemplos los ejemplos](#)
- [Curso de introducción a Scala en CodelyTV Pro](#) (¡Entra dentro del precio de la suscripción!)

Next >

🕒 You earned 100/100pts • 🔄 Reset

Group Discussion

Start a discussion

- VG

Cómo hacerlo en lenguajes "sin promesas" (PHP) ?
Victor G. · a year ago · 🗨️ +1 · +3

Answer >
- FD

Código
Fabian Eduardo D. · a year ago · 🗨️ +0 · +2

Answer >

