02/15/2020

Carlos Cordero

Report

Working Hard to Keep it simple (https://www.youtube.com/watch?v=3jg1AheF4n0)

The speed of the microcontrollers is not increasing, what is increasing is the number of cores.

We have two problem with this, first, we need to compute in parallel to really take advantage of this and the second one is that we have now bunches of concurrency information.

The parallelism generates many not deterministic situations.

Scala in general is an Object Oriented Language and it is quite similar to java that is also an Object Oriented Language but Scala is more compact.

It is also quite easy to make parallel computing easier in Scala than in Java.

The problem is that the compilation is not easy to be parallelized but what can be done is to optimize the languages.

Why Big Data Needs To Be Functional(https://www.youtube.com/watch?v=DFAdLCqDbLQ)

In the actuality, the quantity is too big to handle it in a classic way. Scala provides some ways to deal with this problem.

Another problem is that the structure of the data is changing quite fast and having static architectures makes not possible to handle it.

Machine Learning is an important factor in this entire dilemma. It is required to compute a lot of information to get an answer.

The classic way of computing that involves reading information from a database and to then create an object to the work with it, this process adds a lot of overhead and it is not flexible. It is more flexible if we work directly with the data to create data structures and work with it. The reality that we are facing right now is that we have more data and we need more Data-Driven Programs and less formal Schemas.

The way to handle now the process in having the information use mappers and output of the mappers goes to some reducers that gather the result of the computing, all of this has to be performed in parallel.

One important concept is the collection that is going to be the way to replace the objects and make the parallel programming easier.

Another too powerful tools are going to be Hadoop and Akka. Hadoop is a framework that can support this paradigm. Akka is developed for functional parallel programming.

Fibonacci Iterativo

|  |
| --- |
| def fibIterative( input : Int ) : Int = {  var firstVal = 0  var secondVal = 1  var i = 0  var addition = 0    while( i < input ) {  addition = firstVal + secondVal  firstVal = secondVal  secondVal = addition  i = i + 1  }  return firstVal  } |

Fibonacci Recursivo

|  |
| --- |
| def fibRecursivo( n : Int) : Int = {  if ((0 == n) | (1 == n))  {  return n  }  else  {  return(fibRecursivo( n-1 ) + fibRecursivo( n-2 ))  }  } |