Spark

In [1]:
```
sc
```

Out[1]:

SparkContext

Spark UI (http://147.175.149.68:4040)

Version
`v2.2.0`

Master
`local[2]`

AppName
`pyspark-shell`

In [2]:
```
data = [1, 2, 3, 4, 5]
distData = sc.parallelize(data)
```

In [3]:
```
f = distData.map(lambda x: x % 2 == 0)
f.take(3)
```

Out[3]:
```
[False, True, False]
```

In [4]:
```
f = distData.filter(lambda x: x % 2 == 0)
f.take(5)
```

Out[4]:
```
[2, 4]
```

Hladanie prvocisel

In [5]:
```
# prevzane z https://districtdatalabs.silvrback.com/getting-started-with-spark-in-python

def isprime(n):
    """
    check if integer n is a prime
    """
    # make sure n is a positive integer
    n = abs(int(n))
    # 0 and 1 are not primes
    if n < 2:
        return False
    # 2 is the only even prime number
    if n == 2:
        return True
    # all other even numbers are not primes
    if not n & 1:
        return False
    # range starts with 3 and only needs to go up the square root of n
    # for all odd numbers
    for x in range(3, int(n**0.5)+1, 2):
        if n % x == 0:
            return False
    return True
```

In [6]:
```
import time
```

In [7]:
```
nums = sc.parallelize(range(10**6))
```

In [8]:
```
start = time.time()
print(nums.filter(isprime).count())
end = time.time()
print("Elapsed time: {} s".format(end - start))
```

```
78498
Elapsed time: 3.1132118701934814 s
```

Nedistribuovany vypocet

```
In [9]:  l_nums = sc.parallelize(range(10**6), 1)
```

```
In [10]: start = time.time()
         print(l_nums.filter(isprime).count())
         end = time.time()
         print("Elapsed time: {} s".format(end - start))

         78498
         Elapsed time: 4.602782487869263 s
```

Delayed evaluation

```
In [11]: start = time.time()
         primes = nums.filter(isprime)
         end = time.time()
         print("Elapsed time: {} s".format(end - start))

         Elapsed time: 0.00011873245239257812 s
```

v predchadzajucej bunke sa este nic nevykonalo, nepouzil som funkiu, ktora by mala vracat nejaky vysledok, tak sa len pripravil vypocet na spustenie, ale nespustil sa

```
In [12]: start = time.time()
         # print nums.filter(isprime).take(5)
         print(nums.filter(isprime).takeOrdered(5, key = lambda x: -x))
         end = time.time()
         print("Elapsed time: {} s".format(end - start))

         [999983, 999979, 999961, 999959, 999953]
         Elapsed time: 2.866408586502075 s
```

MapReduce na spracovanie suboru

```
In [13]: # Teraz tie data natahujem z jednueho suboru na disku jedneho fyzickeho pocitaca, ale mohol by som pouzit
         # natiahnutie z dristribuovaneho suoroveho systemu a vykonavat ten vypocet na celom clustri.
         # Jediny rozdiel by bol v tomto riadku
         distFile = sc.textFile("data/shakespeare.txt")
```

```
In [14]: distFile.count()

Out[14]: 147928
```

```
In [15]: distFile.first()

Out[15]: 'Project Gutenberg’s The Complete Works of William Shakespeare, by William'
```

```
In [16]: distFile.take(5)

Out[16]: ['Project Gutenberg’s The Complete Works of William Shakespeare, by William',
          'Shakespeare',
          '',
          'This eBook is for the use of anyone anywhere in the United States and',
          'most other parts of the world at no cost and with almost no restrictions']
```

```
In [17]: tmp = distFile.filter(lambda line: "JULIA" in line)
```

```
In [18]: tmp.count()

Out[18]: 119
```

```
In [19]: tmp.take(5)

Out[19]: ['  JULIA, a lady of Verona, beloved of Proteus',
          "SCENE II. Verona. The garden Of JULIA'S house",
          'Enter JULIA and LUCETTA',
          '  JULIA. But say, Lucetta, now we are alone,',
          '  JULIA. Of all the fair resort of gentlemen']
```

```
In [20]: distFile.map(lambda s: len(s)).reduce(lambda a, b: a + b)

Out[20]: 5546150
```

```
In [21]: distFile.filter(lambda line: "JULIA" in line).map(lambda s: len(s)).reduce(lambda a, b: a + b)
```