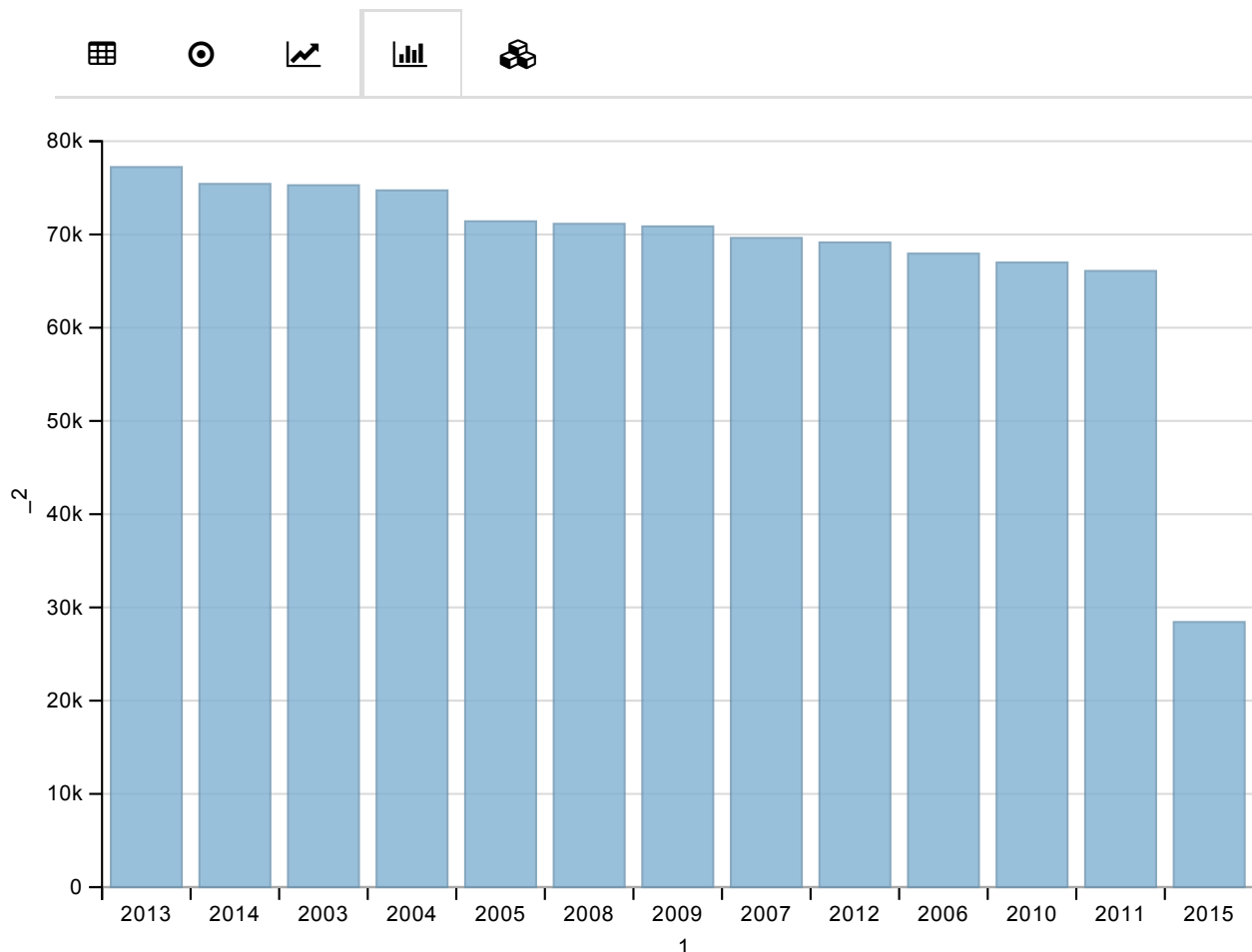


# SFC

```
val rdd = sparkContext.textFile("sfc/test.csv")
val descriptionIndex = rdd.first.split(",").zipWithIndex
val rows = rdd.filter(x => !x.startsWith("Id"))
//rows.first.split(",").zipWithIndex
val samples3 = rows.takeSample(true, 3)
```

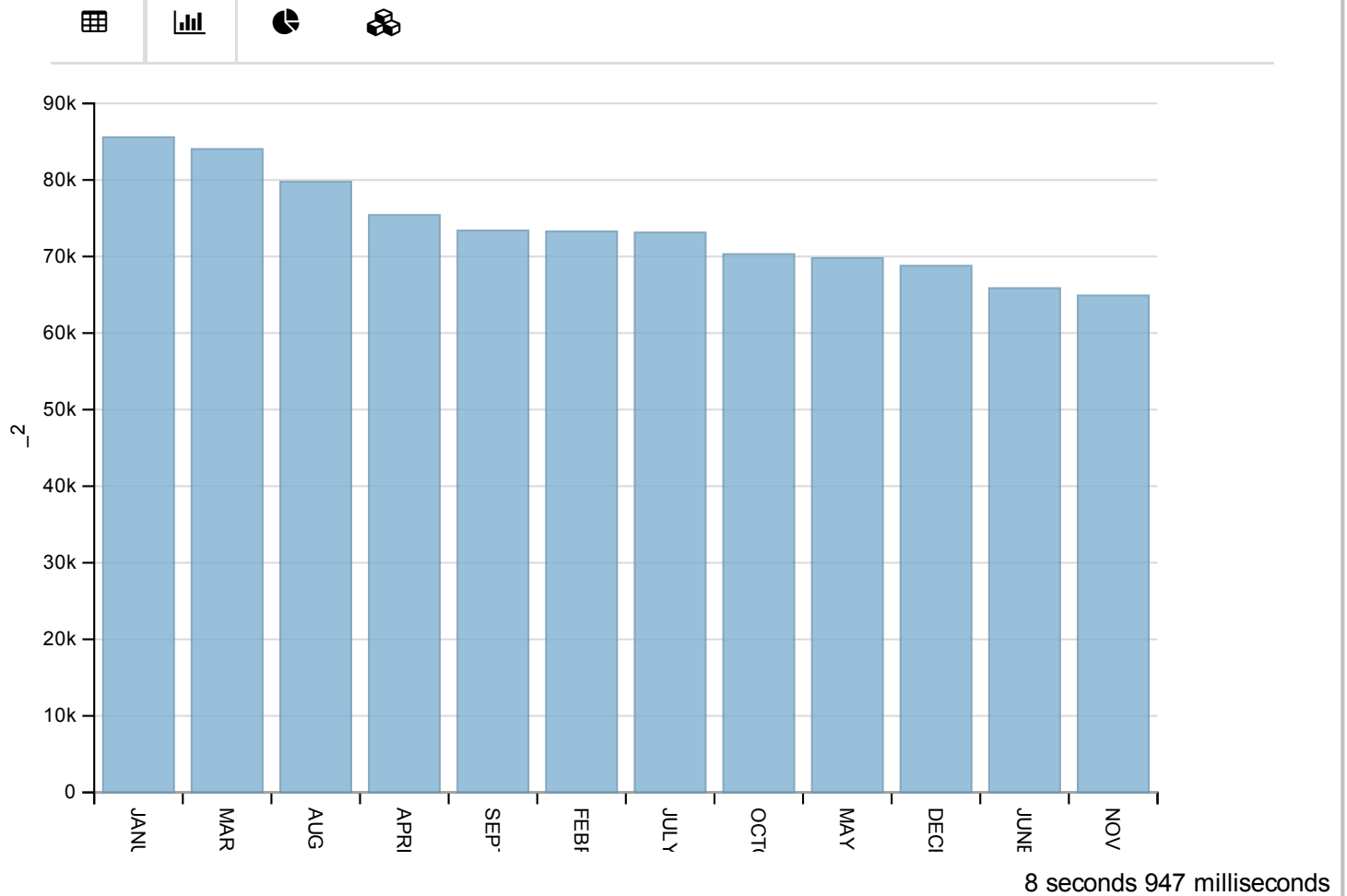
2 seconds 466 milliseconds

```
import java.time._
import java.time.format._
case class StaticRed(year: Int, BRYANT_ST: Int, MARKET_ST: Int, MISSION_ST: Int, POTRERO_ST: Int)
val dates = splits.map(x => LocalDateTime.parse(x(1), DateTimeFormatter.ofPattern("yyyy-MM-dd")).toLocalDate)
dates.groupBy(line => line.getYear).mapValues { x => x.size }.sortBy(x=>x._2, false).collect()
//dates.groupBy(line => line.getYear).mapValues(x => x.size).map(x => x.swap).sortByKey(false)
```

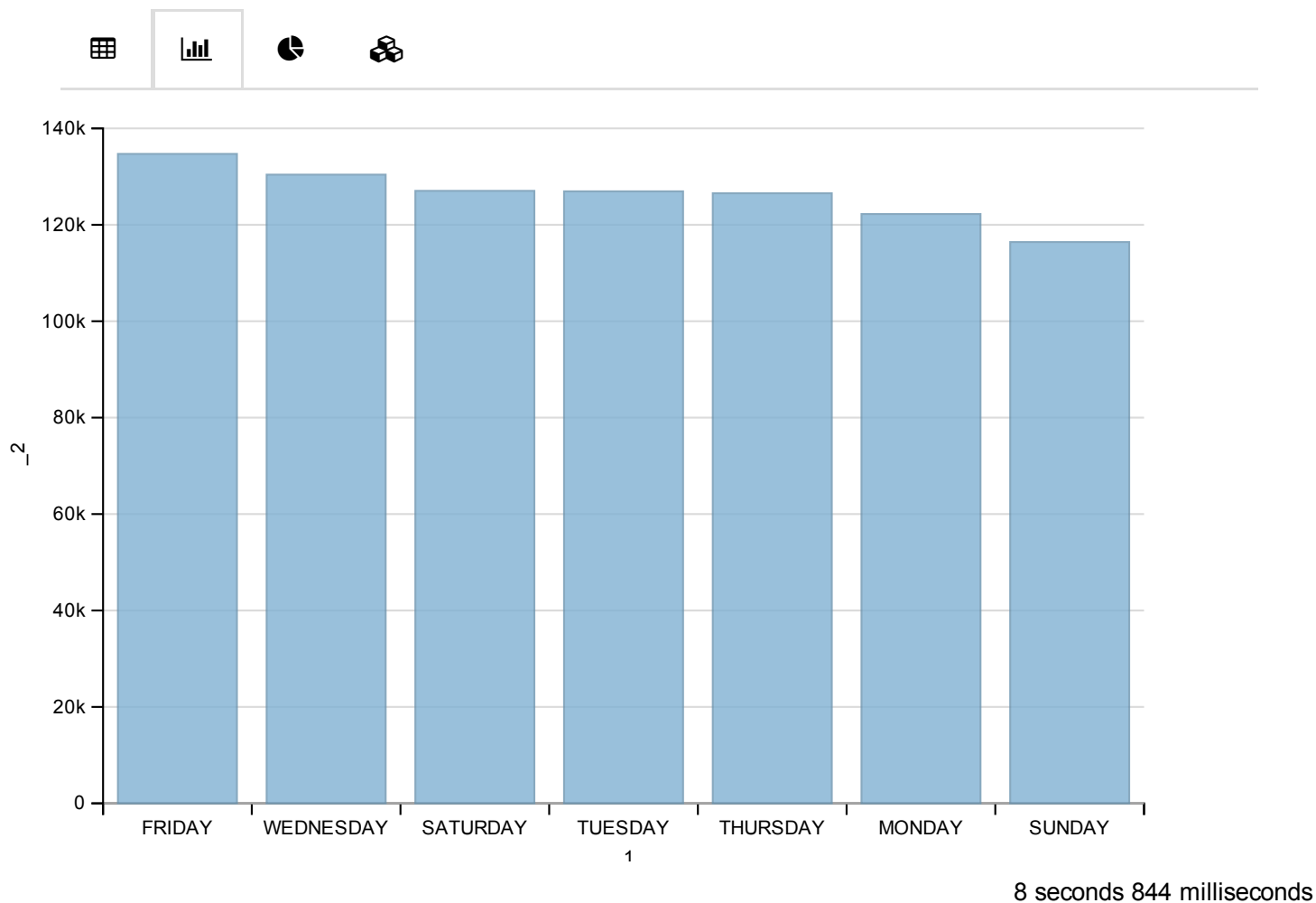


9 seconds 939 milliseconds

```
dates.groupBy(line => line.getMonth).mapValues { x => x.size }.sortBy(x => x._2, false).collect
```



```
dates.groupBy(line => line.getDayOfWeek).mapValues { x => x.size }.sortBy(x => x._2, false)
```



```

case class BlkCrime(blkName: String, crimeNum: Int)
val splits = rows.map(x => x.split(","))
val topCrAdd = splits.map { x => (x(4), 1) }.reduceByKey((x, y) => (x + y)).sortBy(x=>x._2)

val top10CrimeBlk = topCrAdd.map{case (blkName, crimeNum) => BlkCrime(blkName, crimeNum)}

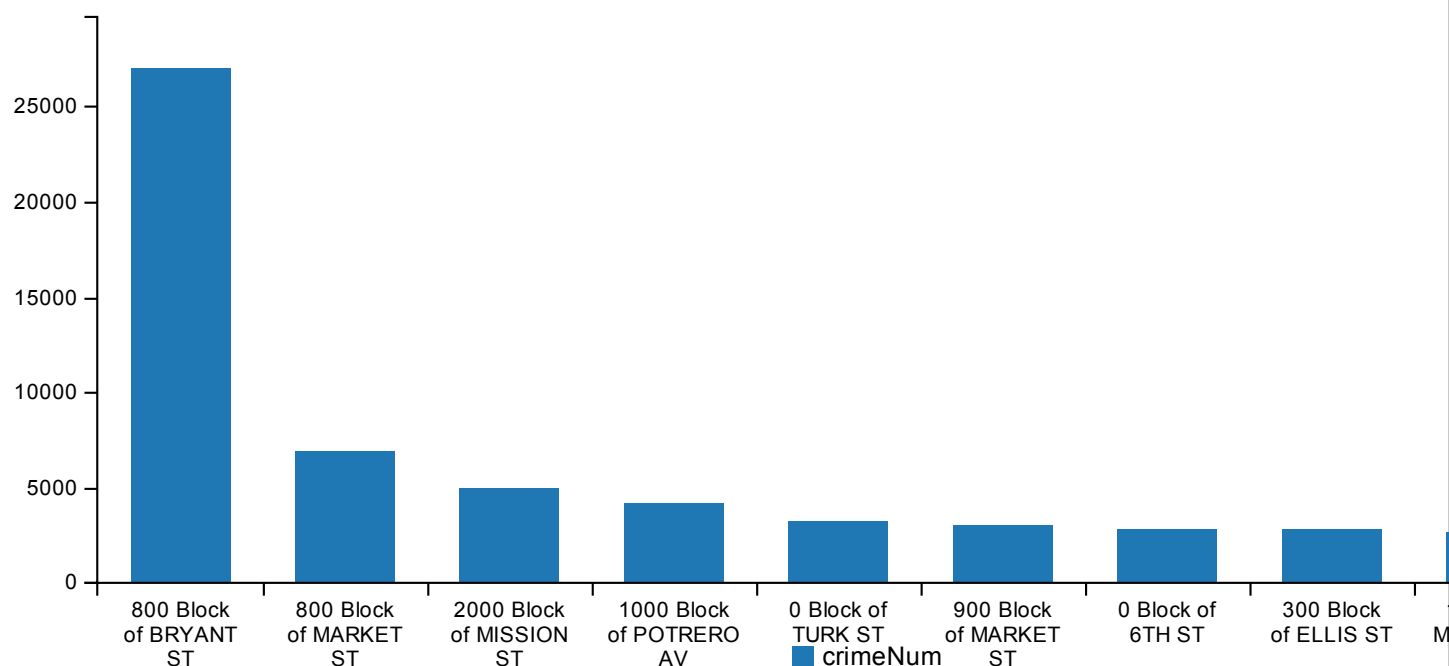
```

3 seconds 270 milliseconds

```

CustomC3Chart(top10CrimeBlk,
  chartOptions = """
    { data: {x: 'blkName', type: 'bar'},
      axis: {x: {type: 'categorical'}}
    }
  """)

```



```

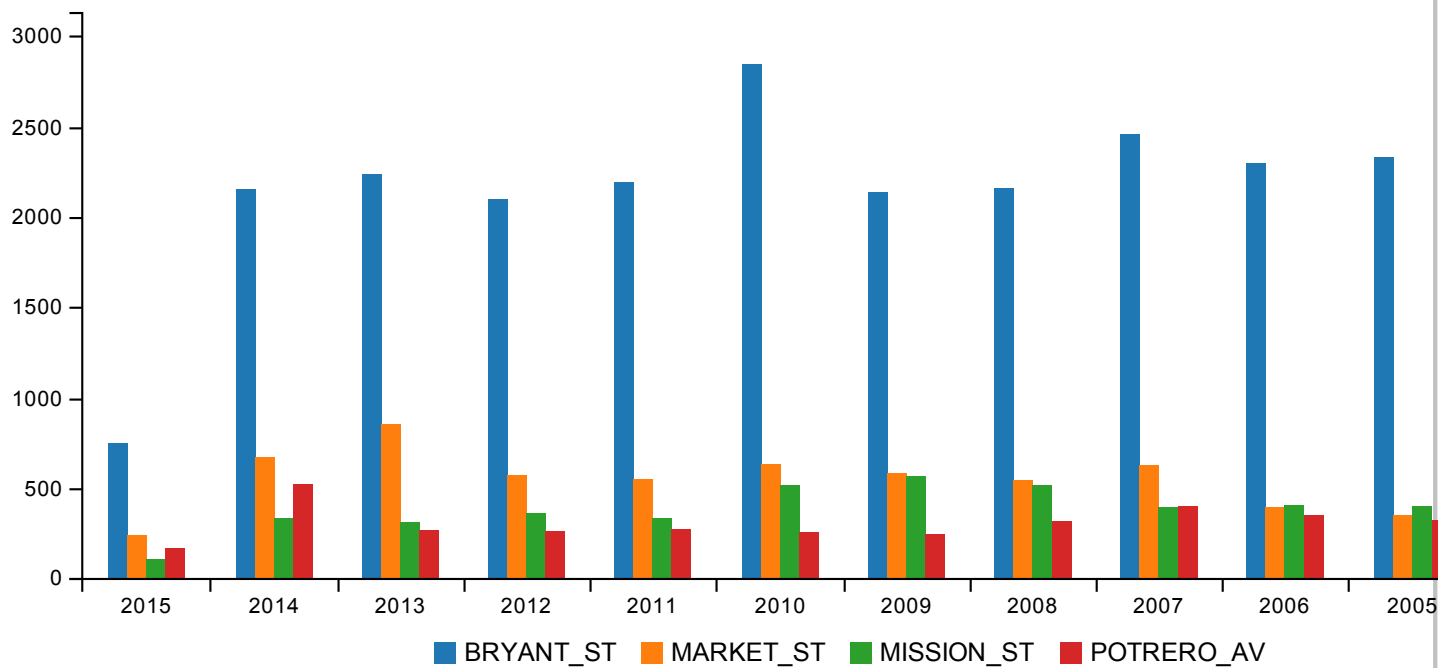
import java.time._
import java.time.format._
case class StaticRed(year: Int, BRYANT_ST: Int, MARKET_ST: Int, MISSION_ST: Int, POTRERO_ST: Int)

val top4AddNames = topCrAdd.map(x => x._1).collect().take(4).toList
val adds = splits.map(x => (x(4).trim,
                           LocalDateTime.parse(x(1), DateTimeFormatter.ofPattern("yyyy-MM-dd HH:mm:ss"))
                           getYear))
//adds.map(x => (x, 1)).reduceByKey((x, y) => (x+y)).map(x => (x._1._1, x._1._2, x._2)).sortByKey(false).mapValues(x => x.toMap)
val top4Add = adds.filter(x => x._1.startsWith(top4AddNames(0)) ||
                           x._1.startsWith(top4AddNames(1)) ||
                           x._1.startsWith(top4AddNames(2)) ||
                           x._1.startsWith(top4AddNames(3)))
.map(x => (x, 1)).reduceByKey((x, y) => (x+y)).map(x => (x._1._2, (x._1._1, x._2)))
.groupByKey().sortByKey(false).mapValues(x => x.toMap)
.map{case (year, map) => StaticRed(year, map(top4AddNames(0)),
                                   map(top4AddNames(1)),
                                   map(top4AddNames(2)),
                                   map(top4AddNames(3)))}.collect
//.map(x => (x._1, x._2.toList(0)._3, x._2.toList(1)._3))
//.map{case (year, list) => StaticRed(year, list.toList(0)._3, list.toList(1)._3)}
//top4Add.take(12)

```

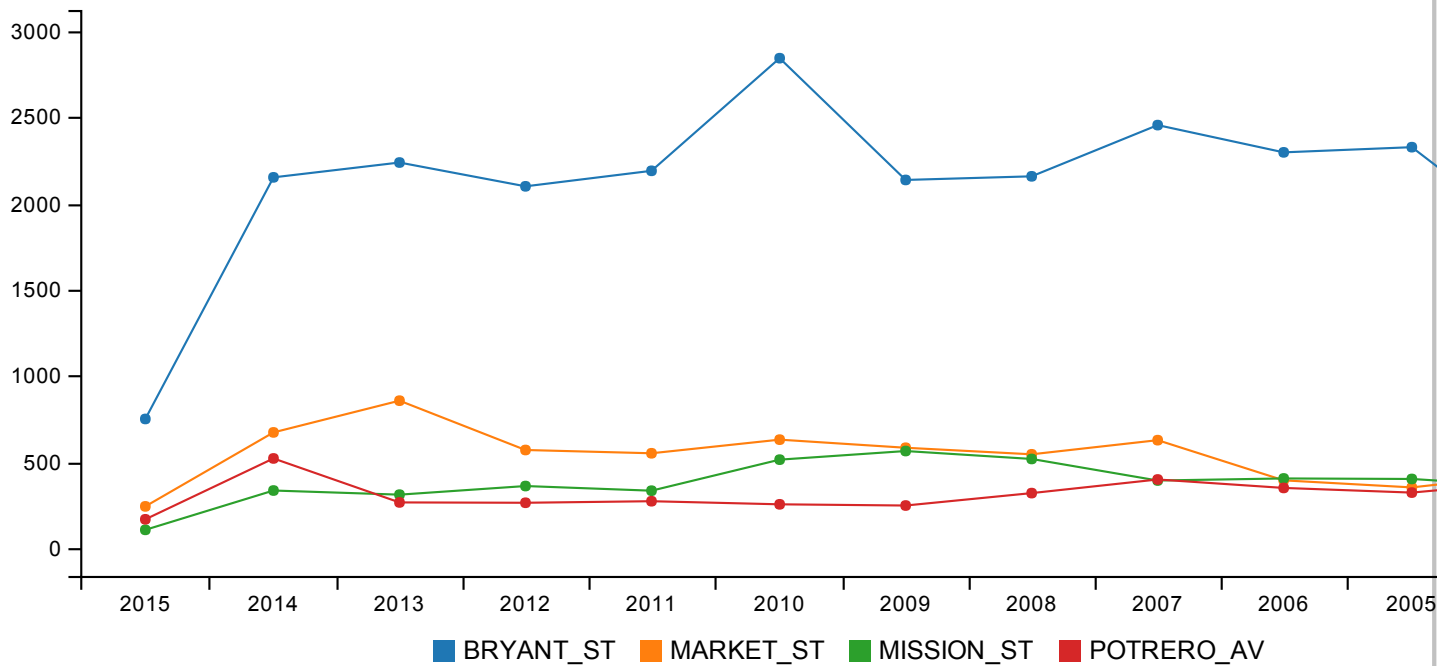
5 seconds 842 milliseconds

```
CustomC3Chart(top4Add,  
  chartOptions = ""  
  { data: {x: 'year', type: 'bar'},  
    axis: {x: {type: 'categorical'}}  
  }  
  """)
```



2 s

```
CustomC3Chart(top4Add,
  chartOptions = ""
  { data: {x: 'year', type: 'line'},
    axis: {x: {type: 'categorical'}}
  }
  "")
```



2 st

Build: | **buildTime**-Thu Mar 17 12:12:20 UTC 2016 | **formattedShaVersion**-0.6.3-61efca56c2a45613ec609f350dc4911b166b3f28-SNAPSHOT | **sbtVersion**-0.13.8 | **scalaVersion**-2.10.5 | **sparkNotebookVersion**-0.6.3 | **hadoopVersion**-2.7.2 | **jets3tVersion**-0.7.1 | **jlineDef**-(org.scala-lang,2.10.5) | **sparkVersion**-1.6.1 | **withHive**-true | **withParquet**-true |.





Spark Job Progress open SparkUI  
(<http://192.168.56.1:4040>)

Terms defined

Name	Type
res3	Array[(String, Int)]
res6	Array[String]
myDateFormat	java.time.format.DateTi
res8	java.time.LocalDateTime

Chat Room

Choose an alias

Type your message...