

IndexedRDD : 高效可更新的Key-value RDD

目前的Spark RDD只提供了一个基于迭代器(iterator-based)、批量更新(bulk-updatable)的接口。但是在很多场景下，我们需要扫描部分RDD便可以查找到我们要的数据，而当前的RDD设计必须扫描全部的分区(partition)。如果你需要更新某个数据，你需要复制整个RDD！那么为了解决这方面的问题，Spark开发团队正在设计一种新的RDD：IndexedRDD。它是一个高效地、基于RDD开发的 key-value store，扩展自RDD[(Long, V)]，保证里面的key是唯一的，为高效的Join操作、点查找、更新以及删除预先建立索引。

RDD：IndexedRDD主要设计包括：

- (1)、基于key对整个数据进行 hash-partitioning；
- (2)、对每个分区内部的hash索引进行维护；
- (3)、用纯粹功能性(不可变以及高效更新)的数据结构来实现高效的更新以及删除等操作。

详细的设计文档可以看这里：

<https://issues.apache.org/jira/secure/attachment/12656374/2014-07-07-IndexedRDD-design-review.pdf>

GraphX 组建将会第一个使用到IndexedRDD，因为它在VertexRDD中实现IndexedRDD了部分的功能，在将来会用IndexedRDD替代VertexRDD。当然，设计者们想到了很多可能会用到IndexedRDD的场景，包括：RDDs的流式更新，direct serving from RDDs，并且可能会作为 Spark SQL的执行策略。

下面是使用IndexedRDD的一个例子：

```
/**
 * User: 过往记忆
 * Date: 15-02-02
 * Time: 上午12:30
 * bolg:
 * 本文地址：/archives/1259
 * 过往记忆博客，专注于hadoop、hive、spark、shark、flume的技术博客，大量的干货
 * 过往记忆博客微信公共帐号：iteblog_hadoop
 */
```

```
import org.apache.spark.rdd.IndexedRDD
```

```
// Create an RDD of key-value pairs with Long keys.
val rdd = sc.parallelize((1 to 1000000).map(x => (x.toLong, 0)))
// Construct an IndexedRDD from the pairs, hash-partitioning and indexing
// the entries.
val indexed = IndexedRDD(rdd).cache()
```

```
// Perform a point update.
```

```
val indexed2 = indexed.put(1234L, 10873).cache()
// Perform a point lookup. Note that the original IndexedRDD remains
// unmodified.
indexed2.get(1234L) // => Some(10873)
indexed.get(1234L) // => Some(0)

// Efficiently join derived IndexedRDD with original.
val indexed3 = indexed.innerJoin(indexed2) { (id, a, b) => b }.filter(_._2 != 0)
indexed3.collect // => Array((1234L, 10873))

// Perform insertions and deletions.
val indexed4 = indexed2.put(-100L, 111).delete(Array(998L, 999L)).cache()
indexed2.get(-100L) // => None
indexed4.get(-100L) // => Some(111)
indexed2.get(999L) // => Some(0)
indexed4.get(999L) // => None
```

目前IndexedRDD 还没有正式发布，相关的代码还在编写中，可能会在spark 1.3.0版本发布。不过如果你想现在使用IndexedRDD，可以加入以下依赖：

```
resolvers += "Sonatype OSS Snapshots" at
  "https://oss.sonatype.org/content/repositories/snapshots"

libraryDependencies += "edu.berkeley.cs.amplab" %% "spark-indexedrdd" % "0.1-SNAPSHOT"
```

并将上述代码片段里面的import org.apache.spark.rdd.IndexedRDD修改成import edu.berkeley.cs.amplab.spark.IndexedRDD

本博客文章除特别声明，全部都是原创！
原创文章版权归过往记忆大数据（[过往记忆](#)）所有，未经许可不得转载。
本文链接: [【】](#) ()