一:SparkSQL与DataFrame

Spark SQL 之所以是除了SparkCore以外最大的和最受关注的组件，原因是：

1. 处理一切存储介质和各种格式的数据（您同时可以方便的扩展SparkSQL的功能来支持更多类型的数据，例如Kudu）；
2. SparkSQL把数据仓库的计算能力推向了新的高度，不仅是无敌的计算速度（Spark SQL比Shark快了至少一个数量级，而Shark比Hive快乐至少一个数量级，尤其是在Tungsten成熟以后会更加无可匹敌），更为重要的是把数据仓库的计算复杂度推向了历史上全新的高度（Spark SQL后续推出了DataFrame可以让 数据仓库直接使用机器学习，图计算等复杂的算法库来对数据仓库进行复杂深度数据价值的挖掘）；
3. SparkSQL（DataFrame、DataSet）不仅是数据仓库的引擎，而且也是数据挖掘的引擎，更为重要的是Spark SQL是数据科学计算和分析引擎；
4. Hive+Spark SQL + DataFrame：
   1. Hive:负责廉价的数据仓库存储；
   2. Spark SQL：负责高速的计算；
   3. DataFrame：负责复杂的数据挖掘；

二：DataFrame与RDD

1. 可以简单的认为Spark中的Dataf是一个分布式的Table

|  |  |  |
| --- | --- | --- |
| Name | Age | Tel |
| Stirng | Int | Long |
| Stirng | Int | Long |
|  | | |
| Stirng | Int | Long |
| Stirng | Int | Long |
| Stirng | Int | Long |

而RDD是形如一下所示：

|  |
| --- |
| Person |
| Person |
| Person |
|  |
| Person |
| Person |
| Person |

1. RDD和DataFrame的根本差异：
   1. RDD是以Record为单位的，Spark在优化的时候无法洞悉Record内部的细节，所以也就无法进行更深度的优化，这极大的限制了Spark SQL性能的提升！
   2. DataFrame包含了每个Record的Metadata信息，也就是说DataFrame的优化是基于列内部的优化，而不是像RDD一样只能够基于行进行优化；