

Spark Tungsten项目的三阶段

基于社区开发者们的观察，绝大多数的Spark应用程序的瓶颈不在于I/O或者网络，而在于CPU和内存。基于这个事实，开发者们发起了Tungsten项目，而Spark 1.5是Tungsten项目的第一阶段。Tungsten项目主要集中在三个方面，于此来提高Spark应用程序的内存和CPU的效率，使得性能能够接近硬件的限制。

Tungsten项目的三个阶段

内存管理和二进制处理（Memory Management and Binary Processing）

- 1、避免使用非transient的Java对象（它们以二进制格式存储），这样可以减少GC的开销。
- 2、通过使用基于内存的密集数据格式，这样可以减少内存的使用情况。
- 3、更好的内存计算（字节的大小），而不是依赖启发式。

4、对于知道数据类型的操作（比如DataFrame和SQL），我们可以直接对二进制格式进行操作，这样我们就不需要进行系列化和反系列化的操作。

缓存感知计算（Cache-aware Computation）

对aggregations, joins和shuffle操作进行快速排序和hash操作。

代码生成（Code Generation）

- 1、更快的表达式求值和DataFrame/SQL操作。
- 2、快速系列化

大多数的Tungsten项目是在DataFrame中进行的，这样使得我们可以获取到关于Application的更多语义。如果有需要，社区还将对Spark的RDD API进行一些改造。下面是Spark 1.5进行的一些关于Tungsten项目的Issues

SPARK-7076 Binary processing compact tuple representation
SPARK-7077 Binary processing hash table for aggregation
SPARK-7080 Binary processing based aggregate operator
SPARK-7081 Faster sort-based shuffle path using binary processing cache-aware sort
SPARK-7082 Binary processing external sort-merge join
SPARK-7083 Binary processing dimensional join
SPARK-7184 Investigate turning codegen on by default
SPARK-7190 UTF8String backed by binary data
SPARK-7251 Perform sequential scan when iterating over entries in BytesToBytesMap

SPARK-7288 Suppress compiler warnings due to use of sun.misc.Unsafe
SPARK-7293 Report memory used in aggregations and joins
SPARK-7311 Enable in-memory serialized map-side shuffle to work with SQL serializers
SPARK-7375 Avoid defensive copying in SQL exchange operator when sort-based shuffle buffers data in serialized form
SPARK-7440 Remove physical Distinct operator in favor of Aggregate
SPARK-7450 Use UNSAFE.getLong() to speed up BitSetMethods#anySet()
SPARK-7517 Rename unsafe module to managedmemory
SPARK-7691 Use type-specific row accessor functions in CatalystTypeConverters' toScala functions
SPARK-7698 Implement buffer pooling / re-use in ExecutorMemoryManager when using HeapAllocator
SPARK-7812 Speed up SQL code generation
SPARK-7813 Push code generation into expression definition
SPARK-7814 Turn code generation on by default
SPARK-7815 Enable UTF8String to work against memory address directly
SPARK-7887 Remove EvaluatedType from SQL Expression
SPARK-7956 Use Janino to compile SQL expression
SPARK-8117 Push codegen into Expression
SPARK-8149 Break ExpressionEvaluationSuite down to multiple files
SPARK-8154 Remove Term/Code type aliases in code generation
SPARK-9700 Pick default page size more intelligently
SPARK-9548 BytesToBytesMap could have a destructive iterator
SPARK-9228 Combine unsafe and codegen into a single option
SPARK-9363 SortMergeJoin operator should support UnsafeRow
SPARK-9693 Reserve a page in all unsafe operators to avoid starving an operator
SPARK-9703 EnsureRequirements should not add unnecessary shuffles when only ordering requirements are unsatisfied
SPARK-8160 Tungsten style external aggregation
SPARK-9736 JoinedRow.anyNull should delegate to the underlying rows
SPARK-7165 Sort-merge Join for left/right outer joins
SPARK-8159 Improve expression function coverage (Spark 1.5)
SPARK-8189 Use 100ns precision for TimestampType
SPARK-8190 ExpressionEvalHelper.checkEvaluation should also run the optimizer version
SPARK-8286 Rewrite UTF8String in Java and move it into unsafe package.
SPARK-8301 Improve UTF8String substring/startsWith/endsWith/contains performance
SPARK-8303 Move DateUtils into unsafe package
SPARK-8305 Improve codegen
SPARK-8307 Improve timestamp from parquet
SPARK-8317 Do not push sort into shuffle in Exchange operator
SPARK-8319 Update logic related to key ordering in shuffle dependencies
SPARK-8354 Fix off-by-factor-of-8 error when allocating scratch space in UnsafeFixedWidthAggregationMap
SPARK-8446 Add helper functions for testing physical SparkPlan operators
SPARK-8498 Fix NullPointerException in error-handling path in UnsafeShuffleWriter
SPARK-8579 Support arbitrary object in UnsafeRow

SPARK-8713 Support codegen for not thread-safe expressions
SPARK-8850 Turn unsafe mode on by default
SPARK-8866 Use 1 microsecond (us) precision for TimestampType
SPARK-8876 Remove InternalRow type alias in expressions package
SPARK-8879 Remove EmptyRow class
SPARK-9022 UnsafeProject
SPARK-9023 UnsafeExchange
SPARK-9024 Unsafe HashJoin
SPARK-9050 Remove out-of-date code in Exchange that was obsoleted by SPARK-8317
SPARK-9054 Rename RowOrdering to InterpretedOrdering and use newOrdering to build orderings
SPARK-9143 Add planner rule for automatically inserting Unsafe Safe row format converters
SPARK-9247 Use BytesToBytesMap in unsafe broadcast join
SPARK-9258 Remove all semi join physical operator
SPARK-9266 Prevent "managed memory leak detected" exception from masking original exception
SPARK-9285 Remove InternalRow's inheritance from Row
SPARK-9329 Bring UnsafeRow up to feature parity with other InternalRow implementations
SPARK-9331 Indent generated code properly when dumping them in debug code or exception mode
SPARK-9334 Remove UnsafeRowConverter in favor of UnsafeProjection
SPARK-9336 Remove all extra JoinedRows
SPARK-9373 Support StructType in Tungsten style Projection
SPARK-9389 Support ArrayType in Tungsten
SPARK-9394 CodeFormatter should handle parentheses
SPARK-9411 Make page size configurable
SPARK-9412 Support records larger than a page size
SPARK-9413 Support MapType in Tungsten
SPARK-9418 Use sort-merge join as the default shuffle join Improvement RESOLVED Reynold Xin
SPARK-9421 Fix null-handling bug in UnsafeRow.getDouble, getFloat(), and get(ordinal, dataType)
SPARK-9448 GenerateUnsafeProjection should not share expressions across instances
SPARK-9450 [INVALID] HashedRelation.get() could return an Iterator[Row] instead of Seq[Row]
SPARK-9457 Sorting improvements
SPARK-9464 Add property-based tests for UTF8String
SPARK-9738 remove FromUnsafe and add its codegen version to GenerateSafe
SPARK-9751 Audit operators to make sure they can support UnsafeRows
SPARK-9784 Exchange.isUnsafe should check whether codegen and unsafe are enabled
SPARK-9785 HashPartitioning compatibility should consider expression ordering
SPARK-9815 Rename PlatformDependent.UNSAFE -> Platform

更多关于Tungsten项目的进展，请关注SPARK-7075和SPARK-9697。

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: **【】** ()