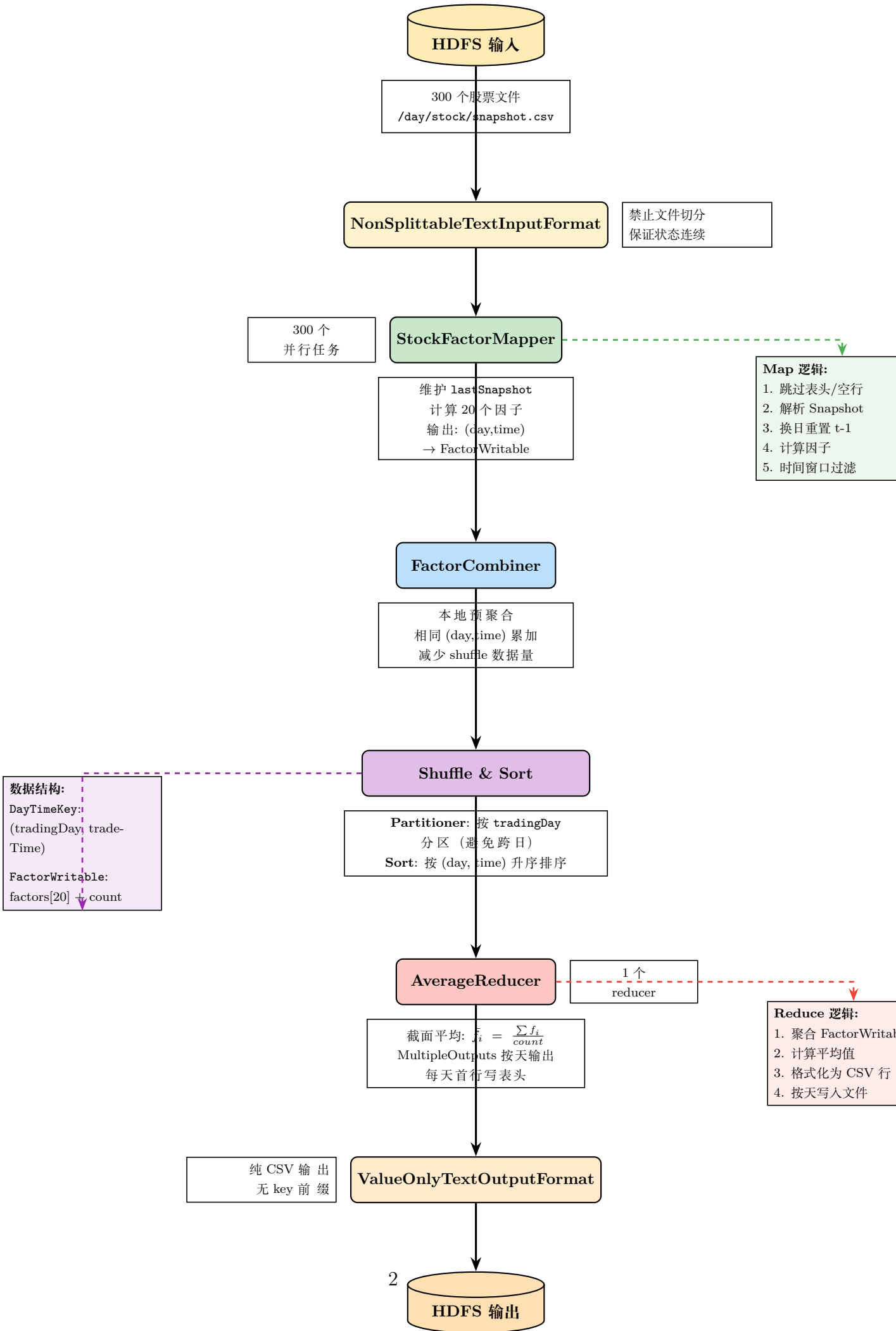
CSI300 因子计算 MapReduce 流水线

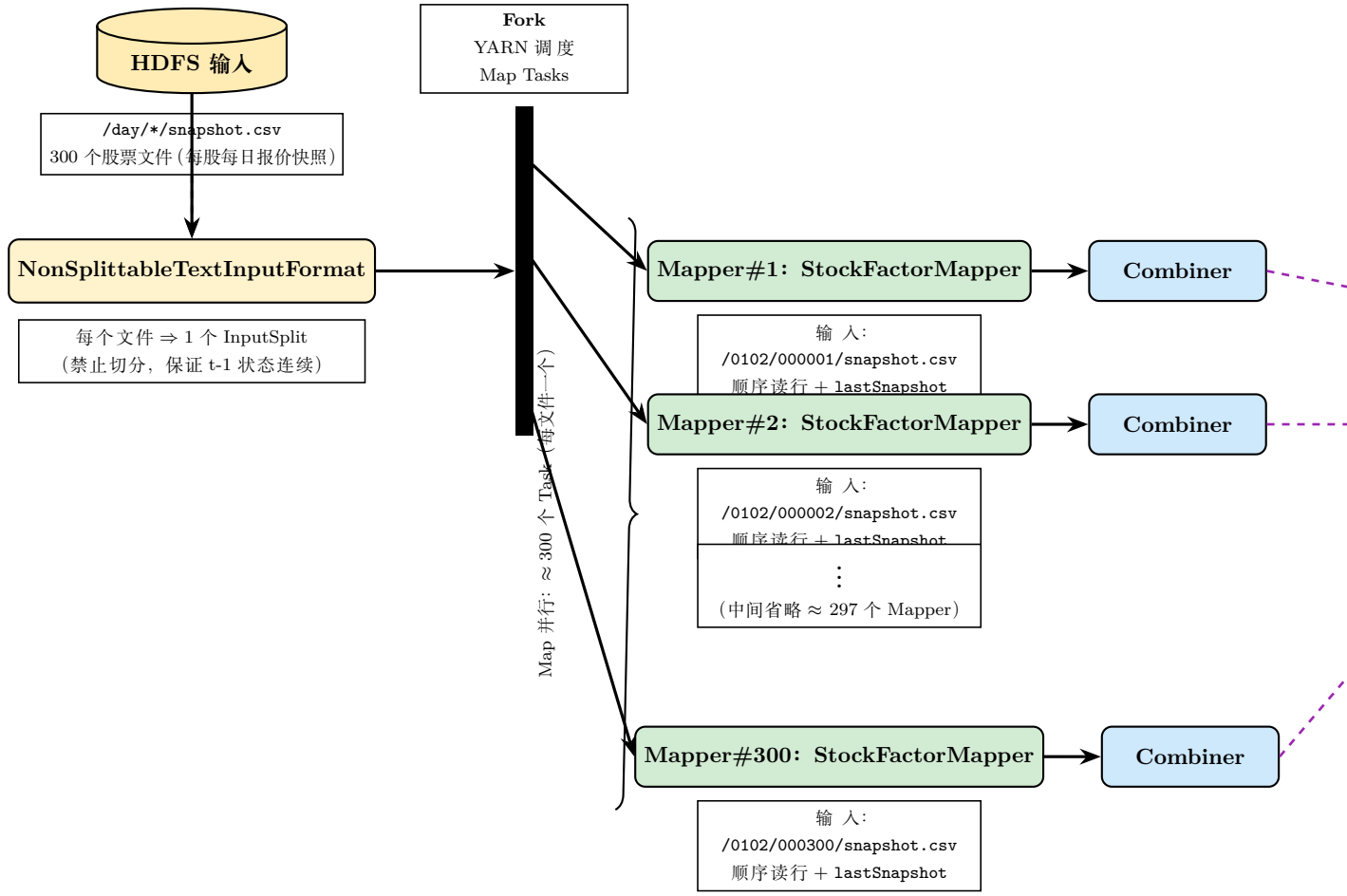
整体架构概览

本项目使用 Hadoop MapReduce 框架，对 CSI300 指数的 300 只股票的高频快照数据（3 秒频）计算 20 个量化因子，并在每个时刻对 300 只股票做截面平均。


HDFS 输入
300 个股票文件
/day/stock/snapshot.csv
NonSplittableTextInputFormat
禁止文件切分
保证状态连续
300 个
并行任务
StockFactorMapper
维护 lastSnapshot
计算 20 个因子
输出: (day,time)
→ FactorWritable
Map 逻辑:
1. 跳过表头/空行
2. 解析 Snapshot
3. 换日重置 t-1
4. 计算因子
5. 时间窗口过滤
FactorCombiner
本地预聚合
相同 (day,time) 累加
减少 shuffle 数据量
Shuffle & Sort
数据结构:
DayTimeKey:
(tradingDay, trade-
Time)
FactorWritable:
factors[20] + count
Partitioner: 按 tradingDay
分区（避免跨日）
Sort: 按 (day, time) 升序排序
AverageReducer
1 个
reducer
截面平均: $\bar{f}_i = \frac{\sum f_i}{count}$
MultipleOutputs 按天输出
每天首行写表头
Reduce 逻辑:
1. 聚合 FactorWritable
2. 计算平均值
3. 格式化为 CSV 行
4. 按天写入文件
纯 CSV 输出
无 key 前缀
ValueOnlyTextOutputFormat
HDFS 输出

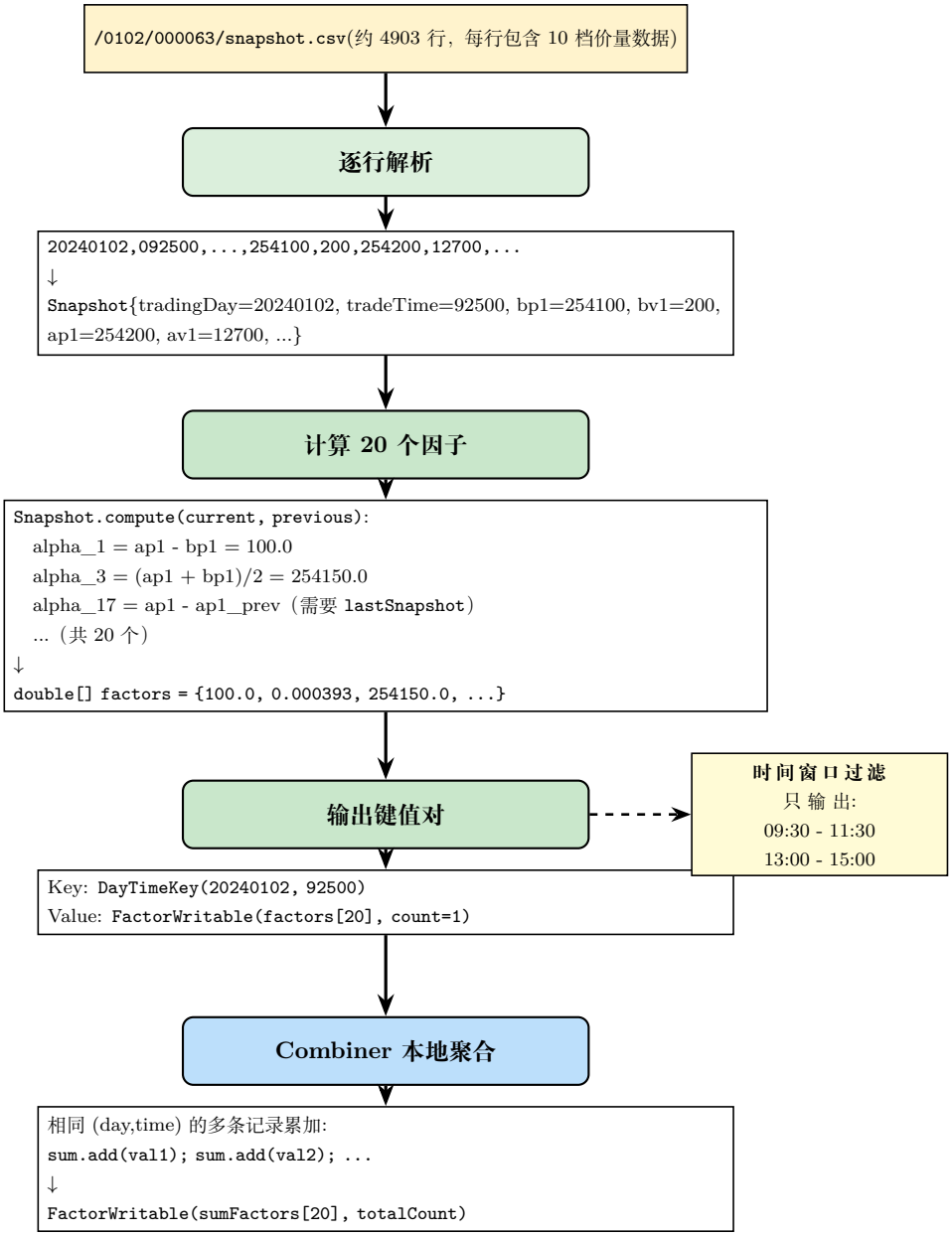
详细数据流示意

1. MapReduce 并行执行示意（Fork/Join）



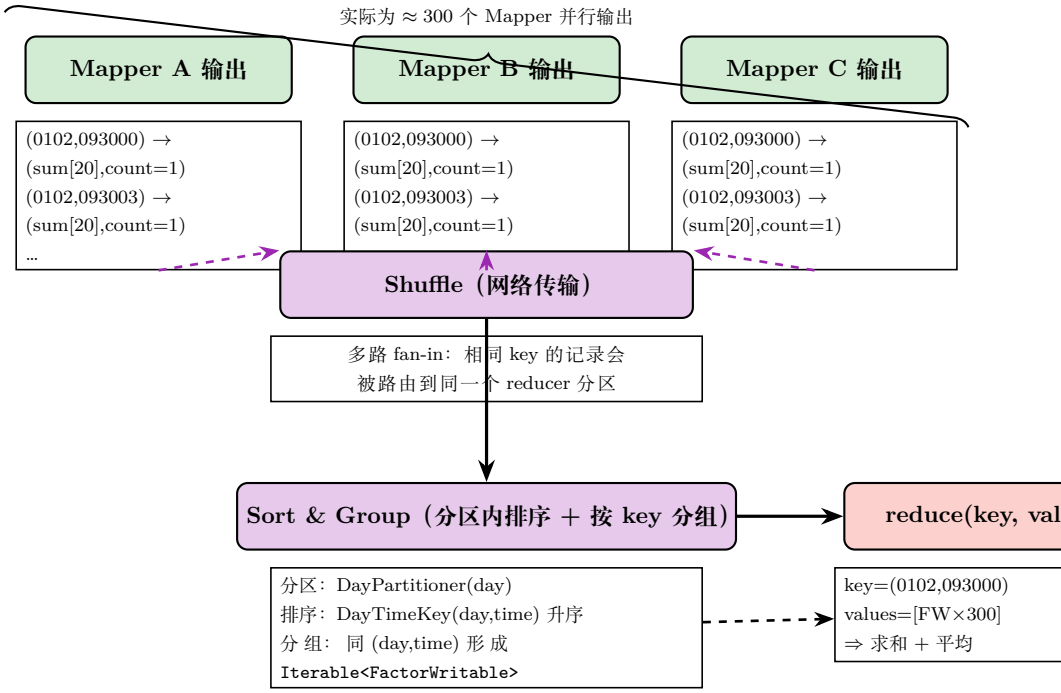
2. 单个 Mapper 的处理流程（顺序读行 + 状态 t-1）

单个 Mapper 的处理流程（以股票 000063 为例）



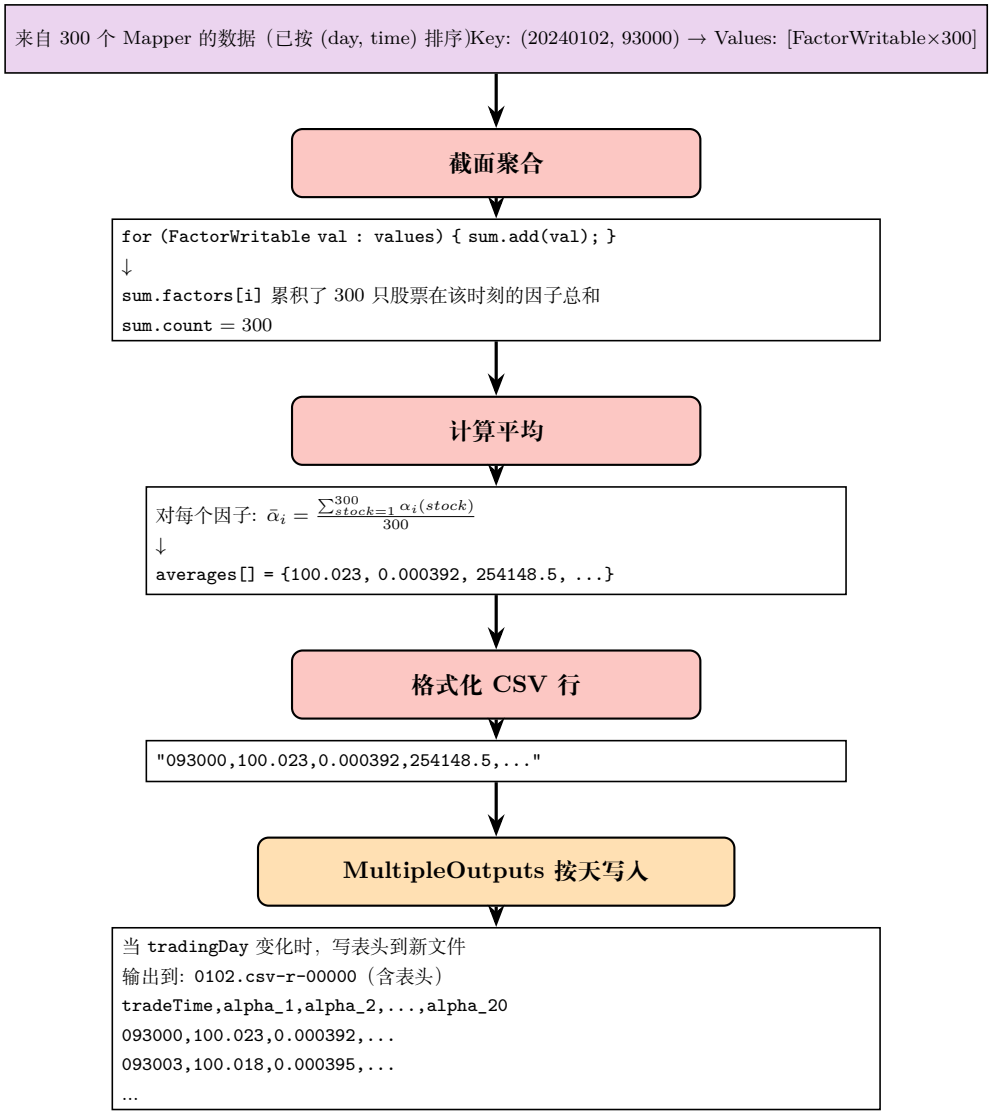
3. Shuffle/Sort：按 (day,time) 分组（Join 前的对齐）

示例：3 个 Mapper（实际 ≈ 300）在同一时刻对齐到同一个 reduce(key)



4. Reducer 聚合与输出（Join：300 股截面平均）

Reducer 聚合与输出流程



关键设计要点

1. 状态维护（Mapper 中的 lastSnapshot）

因子 alpha_17, alpha_18, alpha_19 依赖上一时刻的数据：

$$\alpha_{17} = ap1_t - ap1_{t-1}$$
$$\alpha_{18} = \frac{1}{2}[(ap1_t + bp1_t) - (ap1_{t-1} + bp1_{t-1})]$$
$$\alpha_{19} = \frac{\sum bv_t}{\sum av_t} - \frac{\sum bv_{t-1}}{\sum av_{t-1}}$$

因此 Mapper 必须：

- 维护 `lastSnapshot` 变量
- 按时间顺序处理同一股票的所有记录
- 遇到换日时重置 `lastSnapshot = null`

2. 文件不切分（NonSplittableTextInputFormat）

如果 Hadoop 将一个股票文件切成多个 split：

- Split1 的 Mapper 处理前半部分 ✓
- Split2 的 Mapper 处理后半部分 ×（无法获取 Split1 的最后一条作为 t-1）

解决方案：强制每个文件作为一个整体由单个 Mapper 处理。

3. 按天分区（DayPartitioner）

`DayPartitioner` 只根据 `tradingDay` 计算分区号，确保：

- 同一天的所有数据进入同一个 Reducer
- 不同天的数据不会混算
- Reducer 可以安全地按天输出独立文件

4. 时间窗口过滤（Mapper 中的 shouldEmit）

虽然原始数据包含全天的快照，但标准答案只要求输出交易时段：

- 上午：09:30:00 - 11:30:00
- 下午：13:00:00 - 15:00:00

注意：即使某条记录不在输出窗口，也要维护 `lastSnapshot`，因为 09:30:00 的 t-1 可能来自 09:29:57。

5. 数据流量统计（实际运行结果）

阶段	记录数/数据量
Map input records	1,470,900 行
Map output records	1,440,600 行（过滤后）
Map output bytes	259 MB
Combine input/output	1,440,600 行
Shuffle bytes	263 MB
Reduce input groups	4,802 个 (day,time) 组合
Reduce input records	1,440,600 行
HDFS bytes written	1.76 MB（5 天 CSV）

6. 并行度

- **Mapper 数量**: 300（等于输入文件数，无法改变）
- **Combiner 实例**: 300（每个 Mapper 后自动运行）
- **Reducer 数量**: 1（可通过 `-Dfactor.reducers=N` 调整）

单 Reducer 的优点：

- 每个交易日只写一次表头
- 避免多个 Reducer 输出需要后续合并
- 对于 4802 个时间点，单 Reducer 足够