



图6.3 恢复快照

6.3 CheckPoint

为了保证 State 的容错性, Flink 需要对 State 进行 CheckPoint. CheckPoint 是 Flink 实现容错机制的核心功能, 它能够根据配置周期性地基于 Stream 中各个 Operator/Task 的状态来生成快照, 从而将这些状态数据定期持久化存储下来. Flink 程序一旦意外崩溃, 重新运行程序时可以有选择地从这些快照进行恢复, 从而修正因为故障带来的程序数据异常。

Flink 的 CheckPoint 机制可以与 Stream 和 State 持久化存储交互的前提有以下两点。

- 需要有持久化的 Source, 它需要支持在一定时间内重放事件, 这种 Source 的典型例子就是持久化的消息队列 (如 Apache Kafka、RabbitMQ 等) 或文件系统 (如 HDFS、S3、GFS 等)。
- 需要有用 State 的持久化存储介质, 比如分布式文件系统 (如 HDFS、S3、GFS 等)。

默认情况下, CheckPoint 功能是 Disabled (禁用) 的, 使用时需要先开启它。

通过如下代码即可开启。

```
env.enableCheckpointing(1000);
```

完整的参考代码如下。