

# Streaming Atomic Commits Read Option

## Motivation

We currently do not have a way to enforce **atomic** commit **read** in a streaming micro-batch processing. Either *maxFilesPerTrigger* or *maxBytesPerTrigger* option could end up reading a partial commit if there are multiple actions. We need functionality to read all data or none for a single commit.

We propose adding an option to ensure that each micro-batch reads each commit as a whole.

## Requirements

### **MUST:**

- Do not change current behavior if not specifying this option
- Support atomic commits read when specifying this option
- Respect *maxFilesPerTrigger* / *maxBytesPerTrigger* (softly)
- Not break compatibility with any existing APIs.

### **SHOULD:**

- N/A

### **COULD:**

- Support atomic commits read if not specifying the starting version (read snapshot data)
  - See if we can avoid the initial large batch when reading snapshot

### **Out-of-scope**

- We are only considering Spark Structured Streaming Delta reader here. No effort is made to support continuous mode.

## Proposal sketch

### **OSS Repo and Maven Central**

We will rewrite and add code to the current Delta [project](#).

## Overview

The changes will be mainly on DeltaSource class, adding an option *readAtomicCommits*. We propose that when *readAtomicCommits* is set, DeltaSourceAdmissionControl will make sure actions don't stop in the middle of the commit.

## Implementation

Review and merge [#1079](#) into oss delta