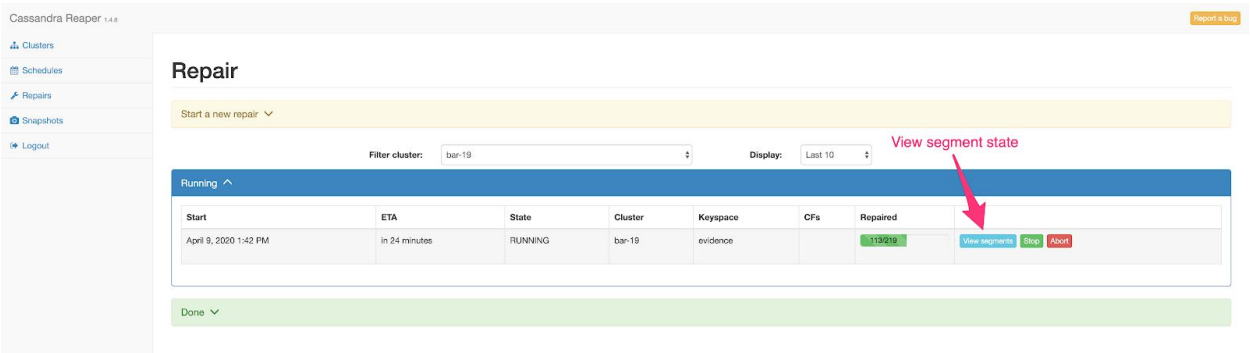
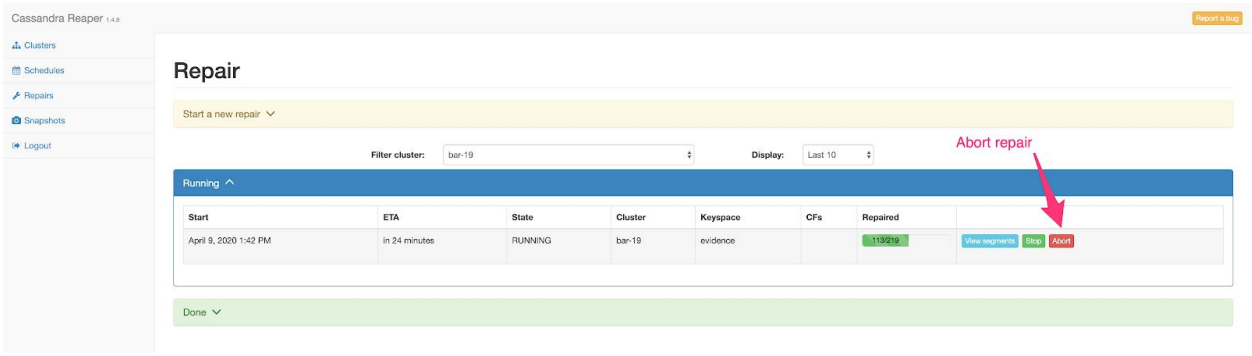
We want to know if the segment is in a `STARTED` state.
Please check the state of segment in the segments view in the Reaper UI.



If the segment is in a `STARTED` state there is a known bug in Reaper where a segment can be in a STARTED state, stalls and never completes. There is a patch coming to fix this in the latest version of Reaper. The state of the segment can be fixed in two possible ways. (edited)

Abort the repair

You can fix this issue by clicking the “Abort” button on the repair with the stalled segment.



This would terminate repairs on any segments that are yet to be completed. You would need to start a new repair which would commence repairing from the start again.

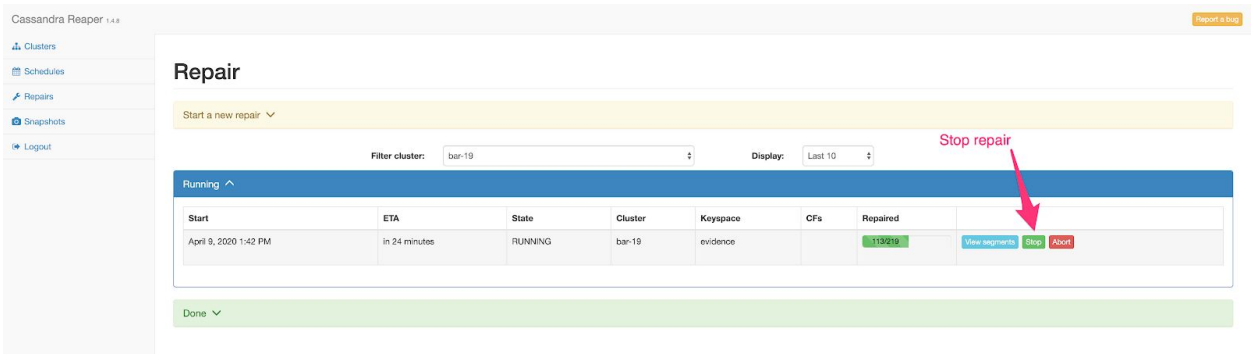
The other way to fix this is more manual.

Modify the entry in the Reaper database

If you wanted to preserve all the work done by the repair so far, you can modify a partition in the `reaper_db.repair_run` table to reset the segment state. This can be done using the following steps.

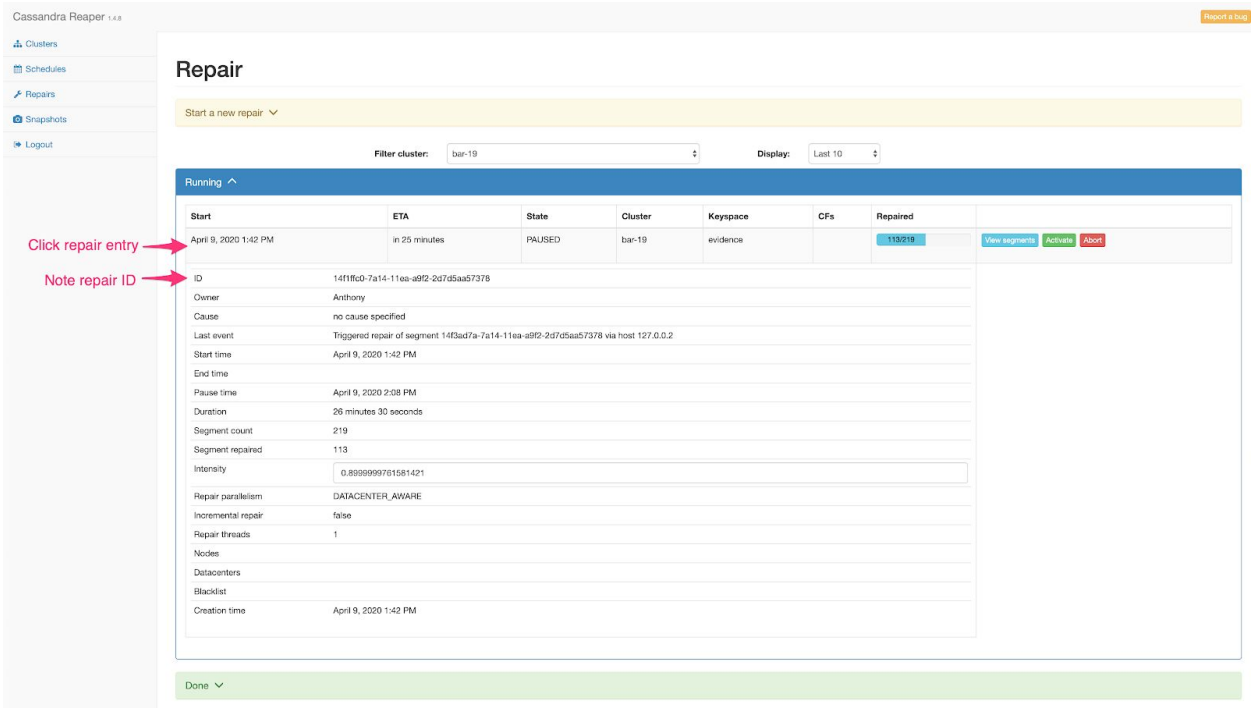
1. Stop the repair

In the Reaper UI stop the repair with the stalled segment by clicking on the “Stop” button.



2. Get the repair ID

In the Reaper UI click on the repair entry with the stalled segment and note the ID of the repair run.



3. Modify the database entry

Open a CQL connection from any Cassandra instance in the cluster. Run the following commands below replacing `<REPAIR_ID>` with the ID recorded in step 2. and `<SEGMENT_ID>` with the segment that fails. Both values must be unquoted.

Confirm we have the correct partition and row in the database, and that segment_state value is greater than 0.

```
cqlsh> SELECT segment_start_time,segment_end_time,segment_state
FROM reaper_db.repair_run
WHERE id = <REPAIR_ID> AND segment_id = <SEGMENT_ID>;
```

Reset the segment state and time.

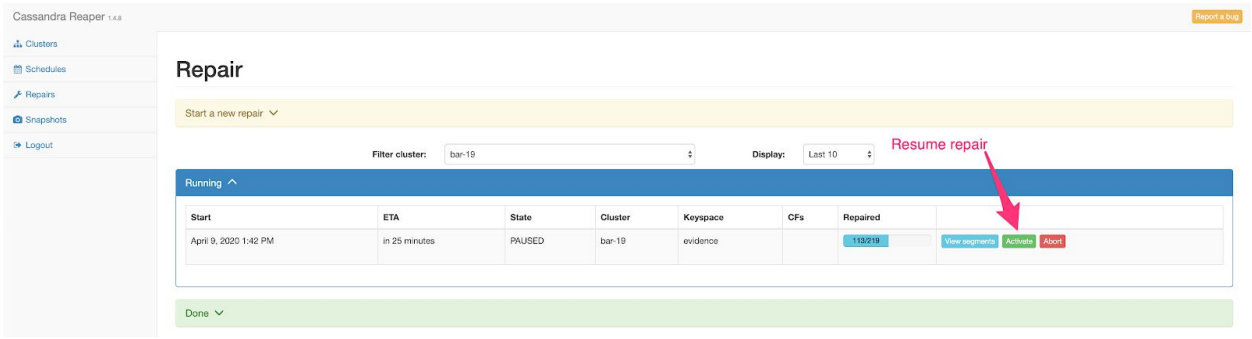
```
UPDATE reaper_db.repair_run
SET segment_state = 0, segment_start_time = null, segment_end_time = null
WHERE id = <REPAIR_ID> AND segment_id = <SEGMENT_ID>;
```

Confirm we have reset the segment row correctly such that its state is 0 and the start/end times are null.

```
cqlsh> SELECT segment_start_time,segment_end_time,segment_state
FROM reaper_db.repair_run
WHERE id = <REPAIR_ID> AND segment_id = <SEGMENT_ID>;
```

4. Start the repair

In the Repair UI start the repair by clicking on the “Activate” button.



If the segment is in another state, i.e. a state other than `STARTED` they will need to supply us with:

- The logs for the Cassandra host
- A screen capture of the segment list for the stalled repair in the Reaper UI

If the segment is continuously postponed, then the node may be participating in another repair. In this case the only option is to restart the Cassandra node.