

SOUTH BAY CASSANDRA USERS MAY 2015

# TTL, TOMBSTONES, AND COMPACTION

Aaron Morton  
@aaronmorton  
Co-Founder & Team Member

**THE LAST PICKLE**

Licensed under a Creative Commons Attribution-NonCommercial 3.0 New Zealand License

About The Last Pickle.

**Work with clients to deliver and  
improve Apache Cassandra based  
solutions.**

**Based in New Zealand, USA,  
Norway.**



# Cassandra Summit 2015

The World's Largest Gathering of Cassandra Users

September 22 - 24, 2015 | Santa Clara, CA

**FREE GENERAL PASSES** Register today!

Visit <http://datastax.com/cassandrasummit2015> for more information



# Storage Engine

## TTL and Tombstones

## Compaction

Log Structured Merge Tree

How to delete data from  
immutable files ?

Tombstones

Use an insert.

Tombstones

**Tombstones create two  
problems.**

Purging The Deleted Data

**Compaction deals with this.**



Purging The Tombstone

`gc_grace_seconds`

“*we may purge the Tombstone  
after this many seconds*”

Distributed Deletes

**Hard deletes leave no  
artefacts.**



# Distributed Deletes

Node	Value	Timestamp
Node 1	null	null
Node 2	null	null
Node 3	foo	10

Distributed Deletes

**Tombstones create and  
artefact of deletion.**



# Distributed Tombstones

Node	Value	Timestamp
Node 1	<Tombstone>	100
Node 2	<Tombstone>	100
Node 3	foo	10

Distributing Tombstones

**Tombstones are distributed by  
AntiEntropy and Read Repair.**



Distributing Tombstones

But we need to set a limit on  
how long they will be  
distributed for.

Distributing Tombstones

`gc_grace_seconds`

“do not distribute tombstones  
older than this many seconds”



# Storage Engine

## **TTL and Tombstones**

### Compaction

TTL

```
public class BufferExpiringCell extends BufferCell
implements ExpiringCell
{
    private final int localExpirationTime;
    private final int timeToLive;
    ...
}
```

# BufferExpiringCell

```
{
    @Override
    public boolean isLive()
    {
        return isLive(System.currentTimeMillis());
    }
    @Override
    public boolean isLive(long now)
    {
        return (int) (now / 1000) < getLocalDeletionTime();
    }
}
```

## Tombstone (For A Cell)

```
public class BufferDeletedCell extends BufferCell implements
DeletedCell
{
    public BufferDeletedCell(CellName name, int localDeletionTime,
long timestamp)
    {
        this(name, ByteBufferUtil.bytes(localDeletionTime),
timestamp);
    }
}
```



# BufferDeletedCell

```
{
    @Override
    public boolean isLive()
    {
        return false;
    }
    @Override
    public boolean isLive(long now)
    {
        return false;
    }
}
```

## Tombstone (For A Partition)

```
public class DeletionInfo implements IMeasurableMemory
{
    public DeletionInfo(long markedForDeleteAt, int
        localDeletionTime)
    {
        ...
    }
}
```

## Tombstone (For A Range)

```
public class RangeTombstone extends Interval<Composite,  
DeletionTime> implements OnDiskAtom  
{  
    public RangeTombstone(Composite start, Composite stop, long  
markedForDeleteAt, int localDeletionTime)  
    {  
        ...  
    }  
}
```

Tombstones In The Read Path

Disk

CQL

Messaging



## Tombstones On Disk

Tombstones are read from SSTables so they can delete live Cells from other SSTables and the Memtable.

Tombstones In CQL

**Tombstones are transformed  
into null when building the  
CQL ResultSet.**

Tombstones In Messaging

**Tombstones are returned to  
the Coordinator or used in  
the Digest calculation.**

# Storage Engine

## TTL and Tombstones

## Compaction



Compaction

Compaction re-writes the  
truth.

# Compaction

Column	SSTable 1	SSTable 2	SSTable 4	New
purple	cromulent (timestamp 10)		<tombstone> (timestamp 15)	<tombstone> (timestamp 15)
monkey		embiggens (timestamp 10, TTL 100)		embiggens (timestamp 10, TTL 100)
dishwasher	tomato (timestamp 10)		tomacco (timestamp 15)	tomacco (timestamp 15)

Compaction

**Purging TTL**

Purging Tombstones

Tombstone Compactions

Expiring SSTables



# Cassandra Summit 2015

The World's Largest Gathering of Cassandra Users

September 22 - 24, 2015 | Santa Clara, CA

**FREE GENERAL PASSES** Register today!

Visit <http://datastax.com/cassandrasummit2015> for more information



Purging TTL

When deserialising  
ExpiringCell create  
DeletedCell if  
localExpirationTime past.



ExpiringCell To DeletedCell

DeletedCell has same  
timestamp as ExpiringCell and  
localDeletionTime set to time  
ExpiringCell was written.

ExpiringCell To DeletedCell

**Normal DeletedCell rules  
now apply.**

ExpiringCell

ExpiringCell has  
localDeletionTime set, so  
looks like a DeletedCell.

Compaction

Purging TTL

**Purging Tombstones**

Tombstone Compactions

Expiring SSTables

Purging Tombstones

Purge the DeletedCell when  
 $\text{localDeletionTime} < \text{gcBefore}$   
and...



Purging Tombstones

there is nothing else for the  
Tombstone to “delete” in non-  
compacting SSTables.

maxPurgeableTimestamp

What is the smallest  
Timestamp for Cells we may  
need to delete in non-  
compacting SSTables?

shouldPurge

We *may* purge the DeletedCell if its timeStamp is less than maxPurgeableTimestamp.

removeDeleted

*We will* purge the DeletedCell  
if it's localDeletionTime is less  
than gcBefore.

Compaction

Purging TTL

Purging Tombstones

**Tombstone Compactions**

Expiring SSTables

## Tombstone Compactions

If we have nothing else to  
compact, look at each\*  
SSTable and...

Tombstone Compactions

**compact it against itself if it  
may free significant space.**



## Tombstone Compactions

If the modified time on  
Data.db is less than  
tombstone\_compaction\_interval  
do nothing.

## Tombstone Compactions

If the estimated purgeable  
Cells ratio is less than  
`tombstone_threshold` do  
nothing.

droppableRatio

**SSTable has a histogram of  
localDeletionTime for all  
Cells.**

(Hint, this includes ExpiringCell)

droppableRatio

Ratio of Cells with  
localDeletionTime before  
gcBefore to total number of  
Cells.

Tombstone Compactions

The SSTable has Tombstones  
to purge, but does it need to  
delete from other SSTables?

Unchecked

If

unchecked\_tombstone\_compaction  
then compact.

Overlapping

If no other SSTables overlap  
the token range of this one  
then compact.



Expired

If this SSTable is *expired* when  
compared to overlaps then  
compact.

Remaining

If estimated ratio of droppable Cells  
that would remain because of  
overlaps greater than  
tombstone\_threshold then  
compact.

Compaction

Purging TTL

Purging Tombstones

Tombstone Compactions

**Expiring SSTables**

Expired SSTable

An SSTable where every Cell  
has a localDeletionTime and...

Expired SSTable

the max Timestamp is less  
than the min Timestamp in any  
SSTable that overlaps.



Expiring In Compaction

**Expired SSTables in a  
compaction task are not  
actually compacted, but are  
deleted.**

Expiring In Compaction

DEBUG

"Dropping expired SSTable {  
(maxLocalDeletionTime={},  
gcBefore={})"

Compaction

Compaction purges TTL and  
Tombstones from disk.

Compaction

Tombstone compactions help  
with data that is NOT actively  
being compacted.

Compaction

Expiring SSTables make  
compaction a meta data  
operation when all Cells are  
TTL or Tombstones.



# Cassandra Summit 2015

The World's Largest Gathering of Cassandra Users

September 22 - 24, 2015 | Santa Clara, CA

**FREE GENERAL PASSES** Register today!

Visit <http://datastax.com/cassandrasummit2015> for more information



Thanks.

Aaron Morton  
@aaronmorton

Co-Founder & Team Member  
[www.thelastpickle.com](http://www.thelastpickle.com)

**THE LAST PICKLE**

Licensed under a Creative Commons Attribution-NonCommercial 3.0 New Zealand License