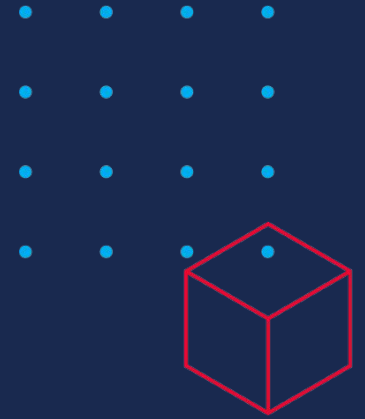
PingCAP TiDB

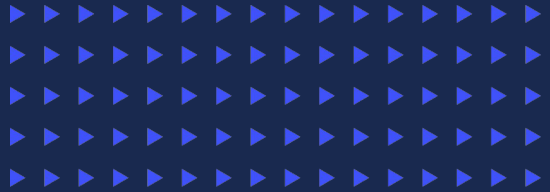


TiFlash DeltaTree Index

Design + Code Sharing (git tag: v6.1.0)

PingCAP

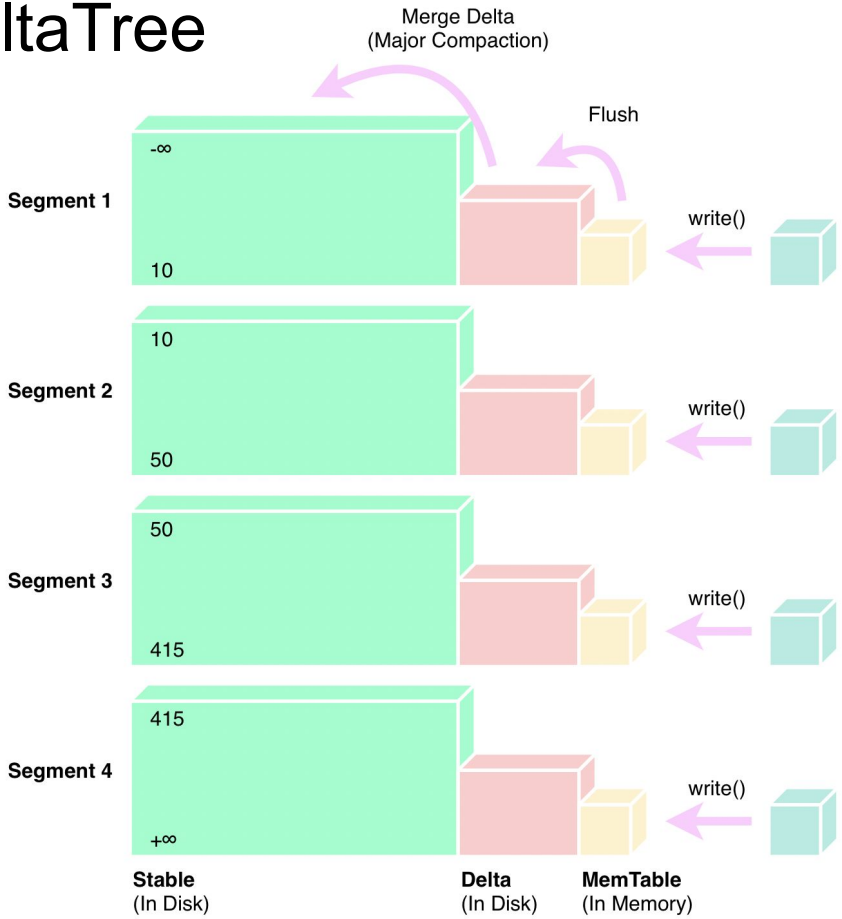
About Me



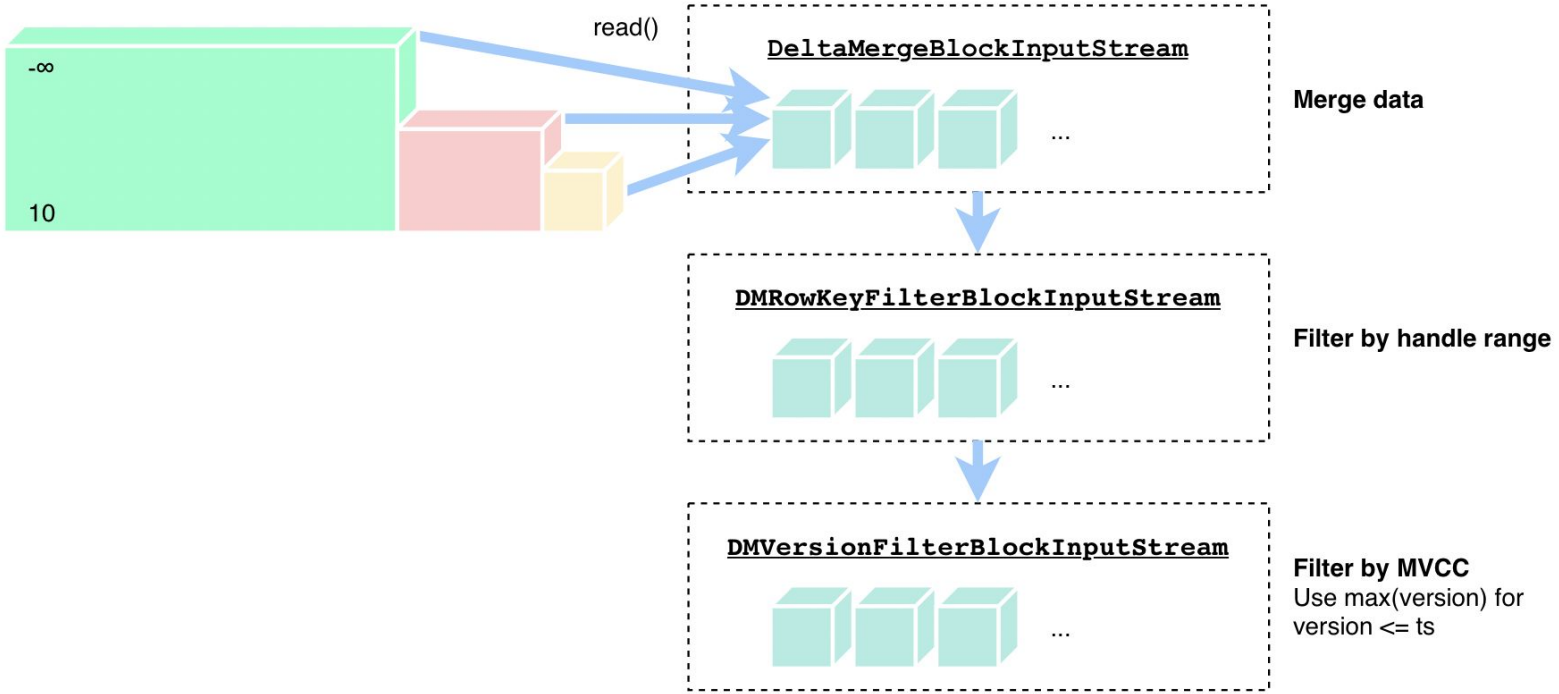
lidezhu https://github.com/lidezhu

TiFlash R&D Engineer

Retrospect: DeltaTree



Retrospect: Scan



K-way Merge?

ColumnFileInMemory[0]

Handle ↑	Version ↑	ColA
Miko	**4**	60

ColumnFileInMemory[1]

Handle ↑	Version ↑	ColA
Albedo	**6**	43
Klee	**8**	80
Klee	**10**	60

MemTable

ColumnFileTiny[0]

Handle ↑	Version ↑	ColA
Klee	**3**	40

ColumnFileTiny[1]

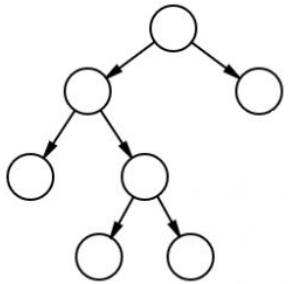
Handle ↑	Version ↑	ColA
Kazuha	**10**	1
Kazuha	**15**	5

Delta

DTFile

Handle ↑	Version ↑	ColA
Albedo	**1**	28
Kazuha	**5**	10
Kazuha	**6**	26
Kazuha	**7**	37
Klee	**1**	23
Klee	**2**	92
Miko	**1**	62
Miko	**7**	13

Stable



Record Merge Process

ColumnFileInMemory[0]

Handle ↑	Version ↑	ColA
Miko	**4**	60

ColumnFileInMemory[1]

Handle ↑	Version ↑	ColA
Albedo	**6**	43
Klee	**8**	80
Klee	**10**	60

MemTable

ColumnFileTiny[0]

Handle ↑	Version ↑	ColA
Klee	**3**	40

ColumnFileTiny[1]

Handle ↑	Version ↑	ColA
Kazuha	**10**	1
Kazuha	**15**	5

Delta

DTFile

Handle ↑	Version ↑	ColA
Albedo	**1**	28
Kazuha	**5**	10
Kazuha	**6**	26
Kazuha	**7**	37
Klee	**1**	23
Klee	**2**	92
Miko	**1**	62
Miko	**7**	13

Stable

read first line of DTFile

read first line of ColmnFileInMemory[1]

read second line of DTFile

……

read eighth line of DTFile

Record Merge Process Using Less Memory

ColumnFileInMemory[0]

Handle ↑	Version ↑	ColA
Miko	**4**	60

ColumnFileInMemory[1]

Handle ↑	Version ↑	ColA
Albedo	**6**	43
Klee	**8**	80
Klee	**10**	60

MemTable

ColumnFileTiny[0]

Handle ↑	Version ↑	ColA
Klee	**3**	40

ColumnFileTiny[1]

Handle ↑	Version ↑	ColA
Kazuha	**10**	1
Kazuha	**15**	5

Delta

DTFile

Handle ↑	Version ↑	ColA
Albedo	**1**	28
Kazuha	**5**	10
Kazuha	**6**	26
Kazuha	**7**	37
Klee	**1**	23
Klee	**2**	92
Miko	**1**	62
Miko	**7**	13

Stable

~~read first line of DTFile~~

read first line of ColmnFileInMemory[1]

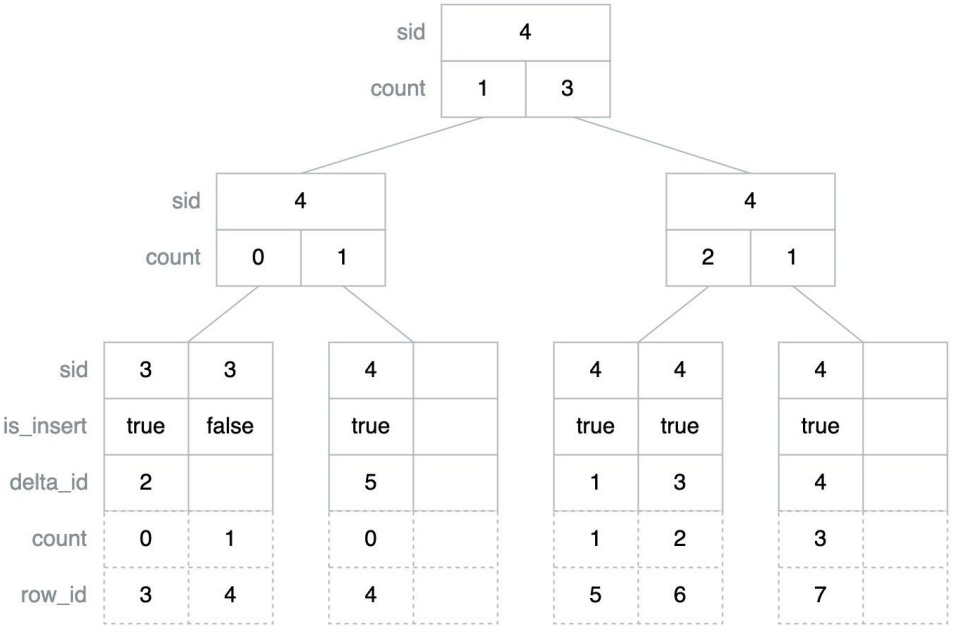
after read one row from stable

~~read second line of DTFile~~

……

~~read eighth line of DTFile~~

Delta Index: B+ Tree-like Structure



sid:
  internal: the minmum sid of right sub tree
  leaf: number of stable rows before this entry

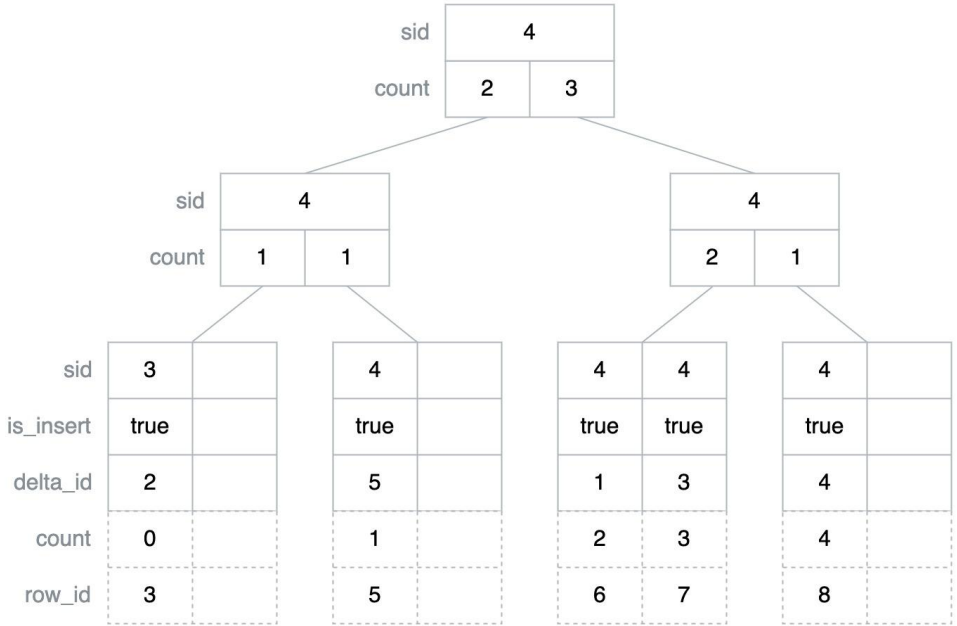
is_insert: whether this entry is insert or delete

count:
  internal: (number of insert - number of delete) in the subtree
  leaf: (number of insert - number of delete) before this entry

delta_id: the corresponding row offset in the delta value space

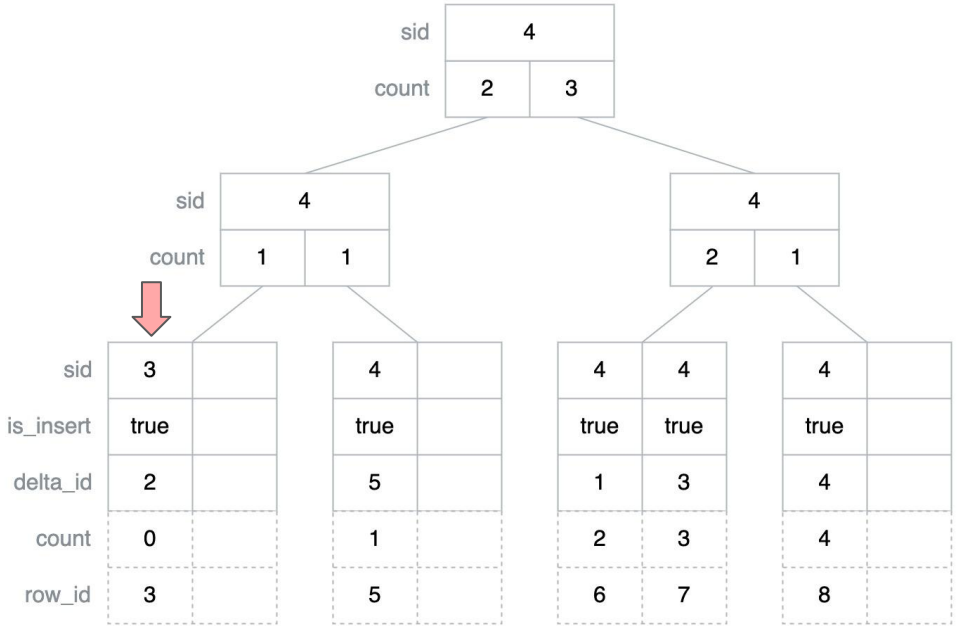
row_id: the actual row id of the row in the final merged stream

Delta Index: Read




```
total_stable_rows = 0
iter = index.begin()
while iter != index.end() {
    rows = iter->sid - total_stable_rows
    read_stable_rows(rows)
    read_delta_row(delta_id)
    total_stable_rows += stable_rows
    iter++
}
```

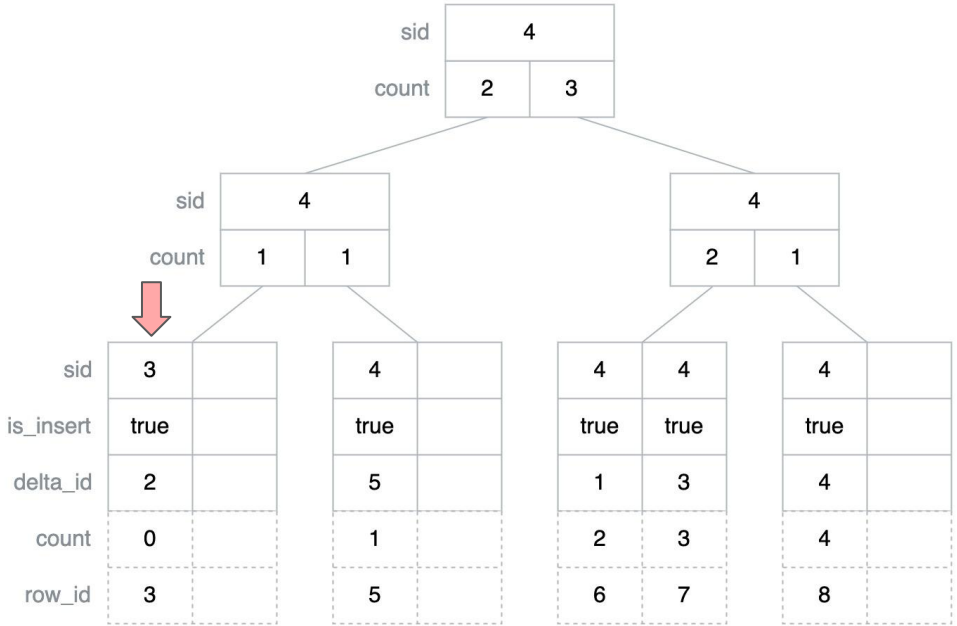
Delta Index: Read




```
total_stable_rows = 0
iter = index.begin()
while iter != index.end() {
    rows = iter->sid - total_stable_rows
    read_stable_rows(rows)
    read_delta_row(delta_id)
    total_stable_rows += stable_rows
    iter++
}
```

read 3 rows from stable

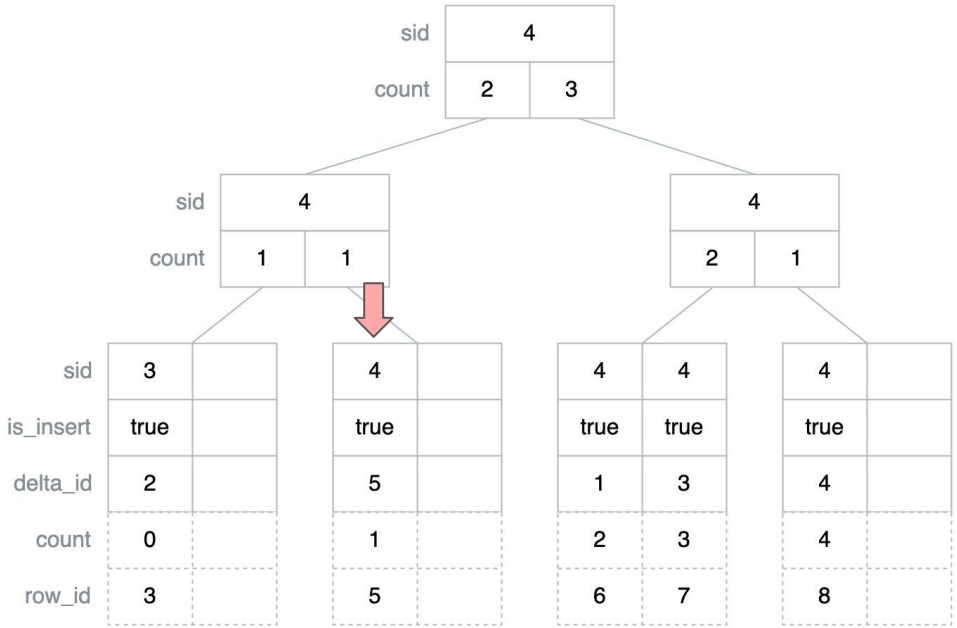
Delta Index: Read




```
total_stable_rows = 0
iter = index.begin()
while iter != index.end() {
    rows = iter->sid - total_stable_rows
    read_stable_rows(rows)
    read_delta_row(delta_id)
    total_stable_rows += stable_rows
    iter++
}
```

read 3 rows from stable
read 1 row from delta at offset 2

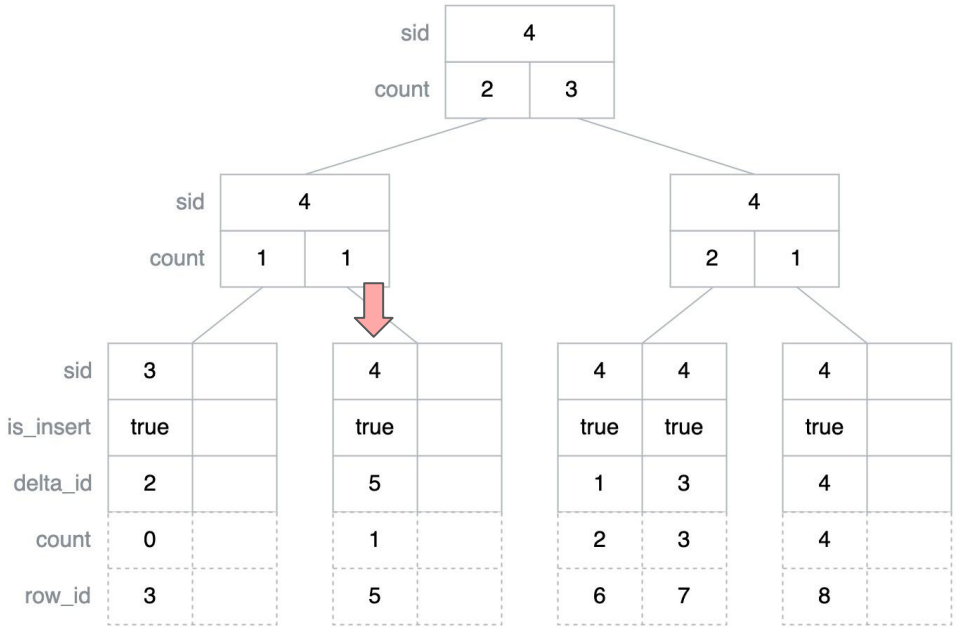
Delta Index: Read




```
total_stable_rows = 0
iter = index.begin()
while iter != index.end() {
    rows = iter->sid - total_stable_rows
    read_stable_rows(rows)
    read_delta_row(delta_id)
    total_stable_rows += stable_rows
    iter++
}
```

read 3 rows from stable
read 1 row from delta at offset 2
read 1 row from stable

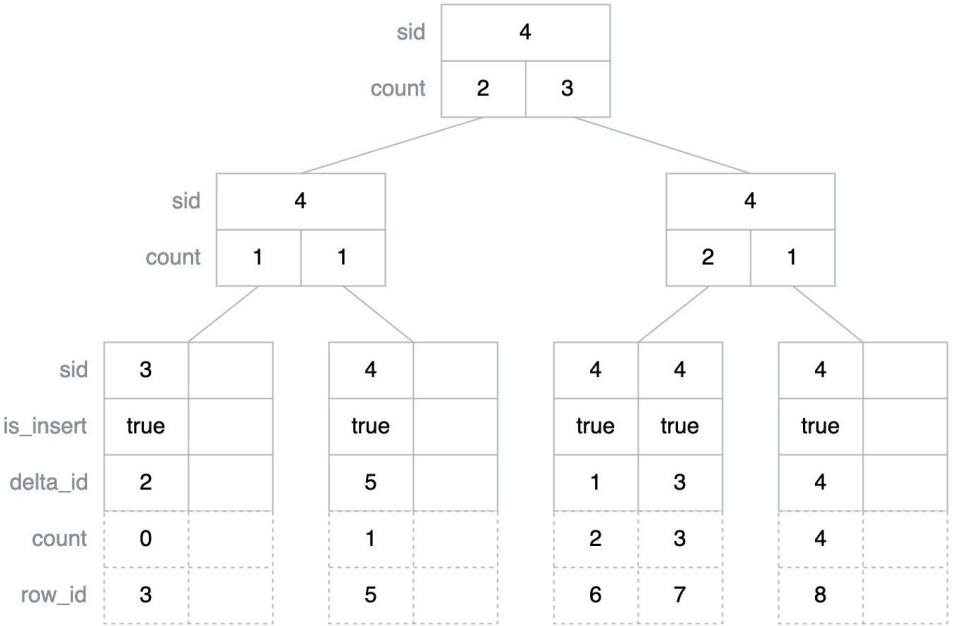
Delta Index: Read




```
total_stable_rows = 0
iter = index.begin()
while iter != index.end() {
    rows = iter->sid - total_stable_rows
    read_stable_rows(rows)
    read_delta_row(delta_id)
    total_stable_rows += stable_rows
    iter++
}
```

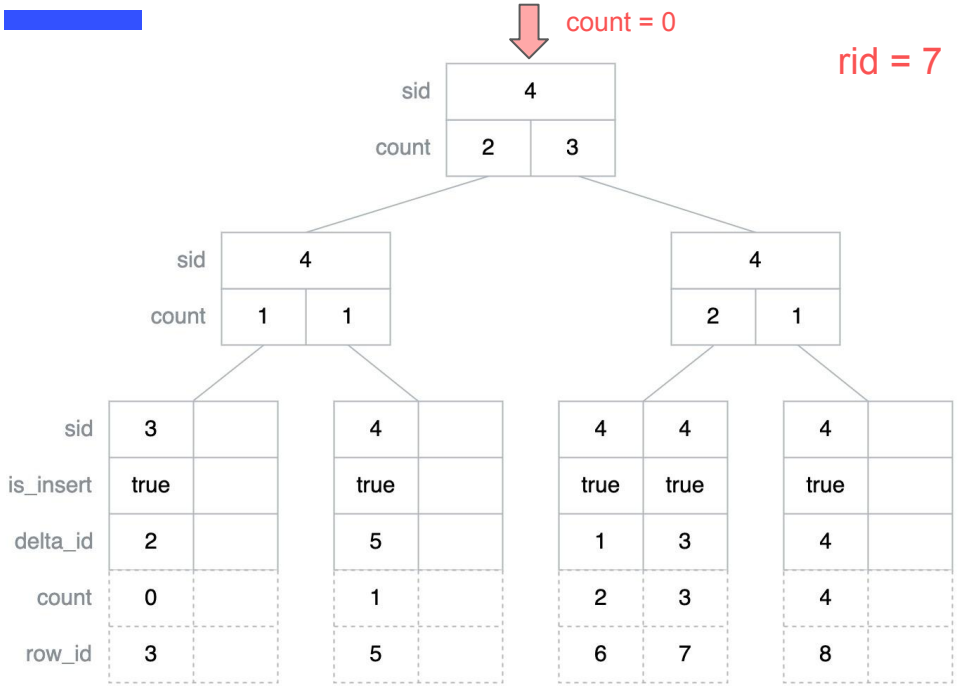
read 3 rows from stable
read 1 row from delta at offset 2
read 1 row from stable
read 1 row from delta at offset 5
……

Delta Index: Search



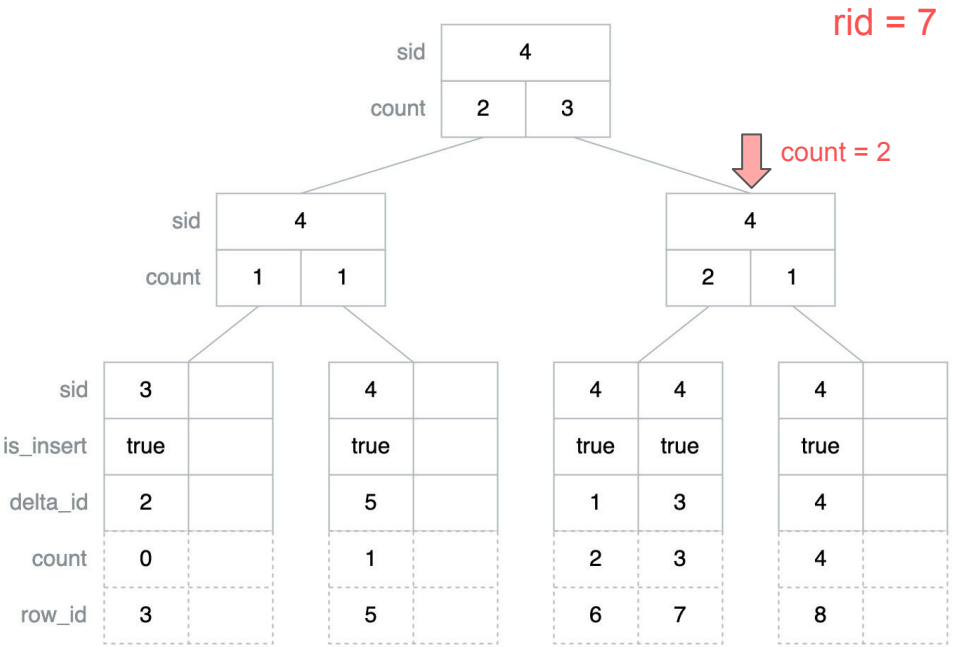

```
findRightLeafByRId(rid) {
    node = root
    count = 0
    while !isLeaf(node) {
        for i = 0; i < child; i++ {
            count = count + node[i].count
            if node[i].sid + count > rid {
                count = count - node[i].count
                break
            }
        }
        node = node[i].child
    }
    return node
}
```

Delta Index: Search



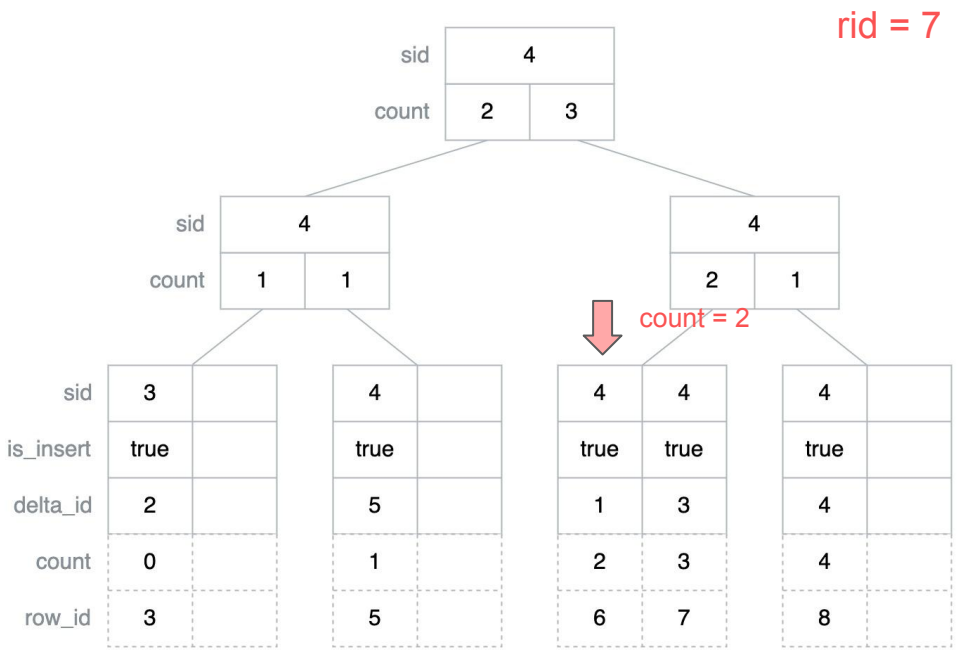

```
findRightLeafByRId(rid) {
    node = root
    count = 0
    while !isLeaf(node) {
        for i = 0; i < child; i++ {
            count = count + node[i].count
            if node[i].sid + count > rid {
                count = count - node[i].count
                break
            }
        }
        node = node[i].child
    }
    return node
}
```

Delta Index: Search



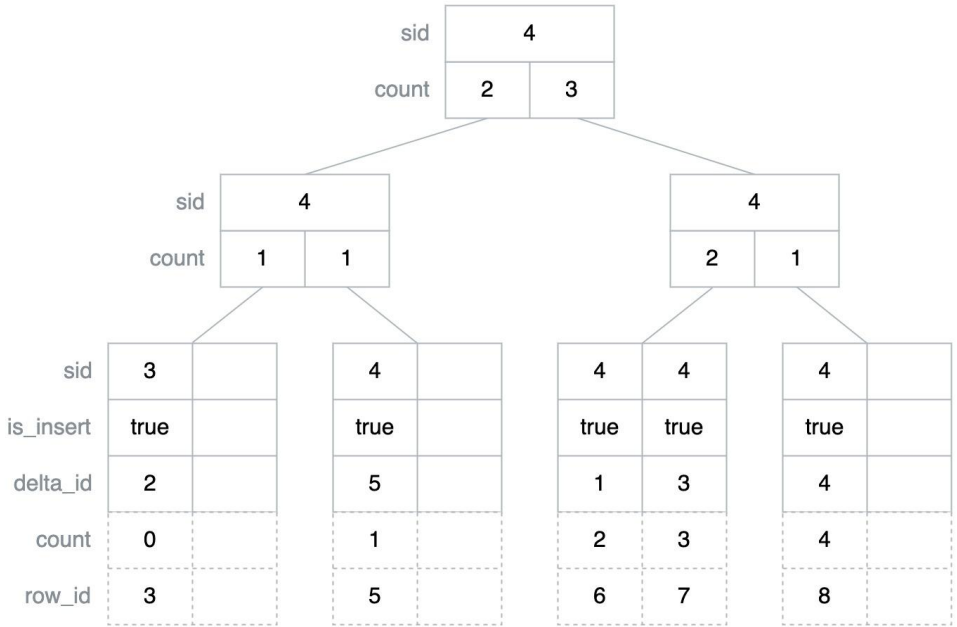

```
findRightLeafByRId(rid) {
    node = root
    count = 0
    while !isLeaf(node) {
        for i = 0; i < child; i++ {
            count = count + node[i].count
            if node[i].sid + count > rid {
                count = count - node[i].count
                break
            }
        }
        node = node[i].child
    }
    return node
}
```

Delta Index: Search



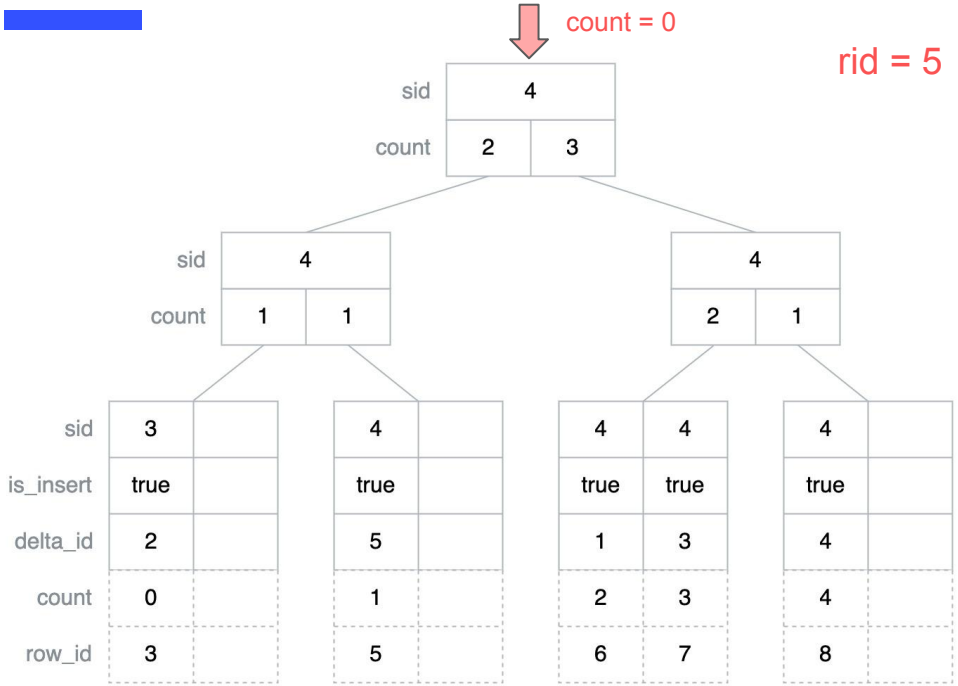

```
findRightLeafByRId(rid) {
    node = root
    count = 0
    while !isLeaf(node) {
        for i = 0; i < child; i++ {
            count = count + node[i].count
            if node[i].sid + count > rid {
                count = count - node[i].count
                break
            }
        }
        node = node[i].child
    }
    return node
}
```

Delta Index: Add Insert




```
leaf, count = findRightLeafByRId(rid)
pos, count = searchLeafForRId(leaf, rid, count)
shiftLeafEntries(leaf, pos, 1)
leaf[pos].sid = rid - count
leaf[pos].delta_id = offset_in_delta_value_space
```

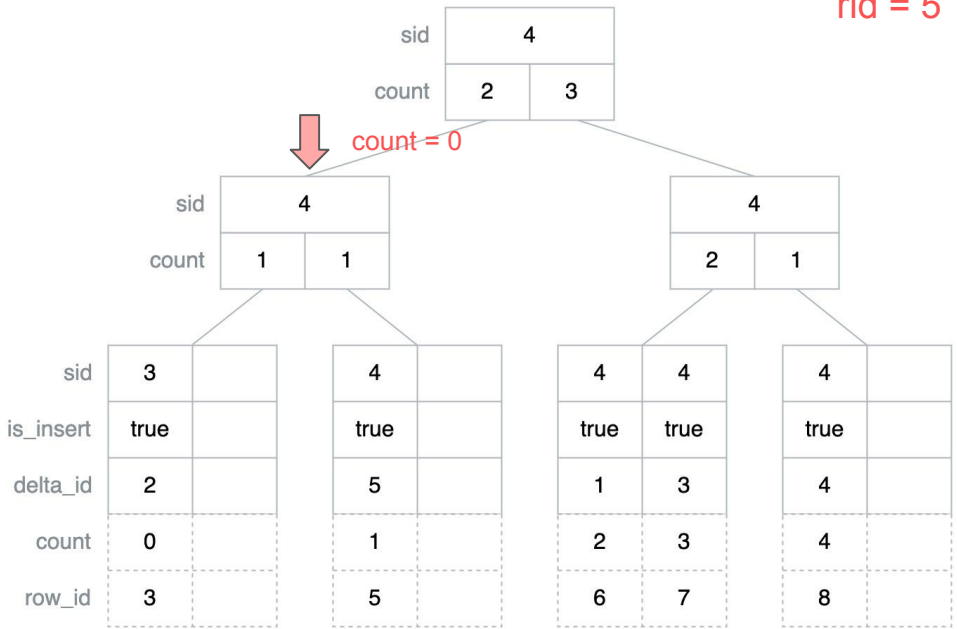
Delta Index: Add Insert



rid = 5

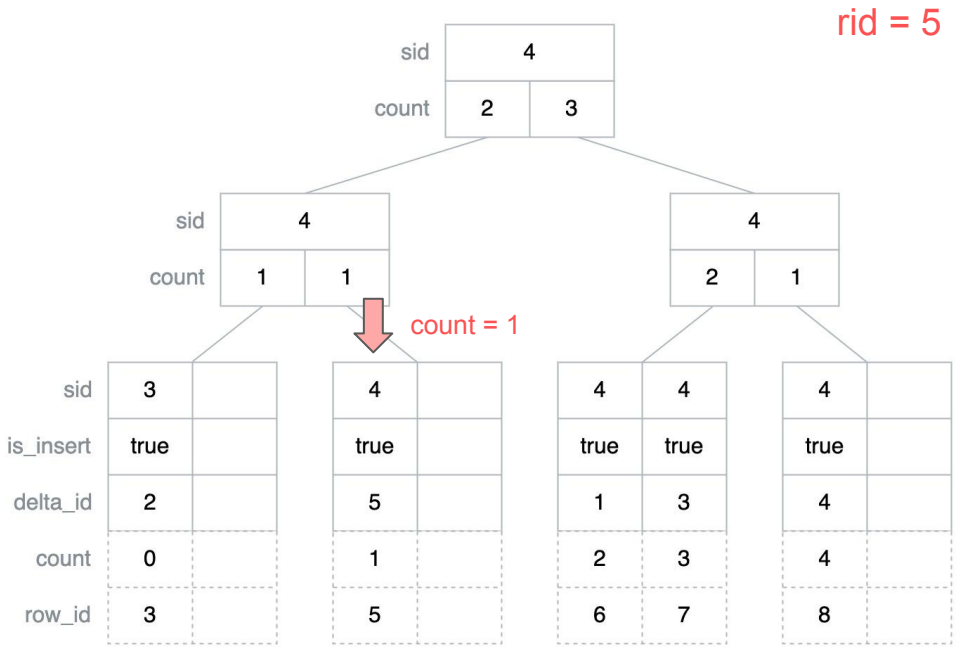
```
leaf, count = findRightLeafByRId(rid)
pos, count = searchLeafForRId(leaf, rid, count)
shiftLeafEntries(leaf, pos, 1)
leaf[pos].sid = rid - count
leaf[pos].delta_id = offset_in_delta_value_space
```

Delta Index: Add Insert



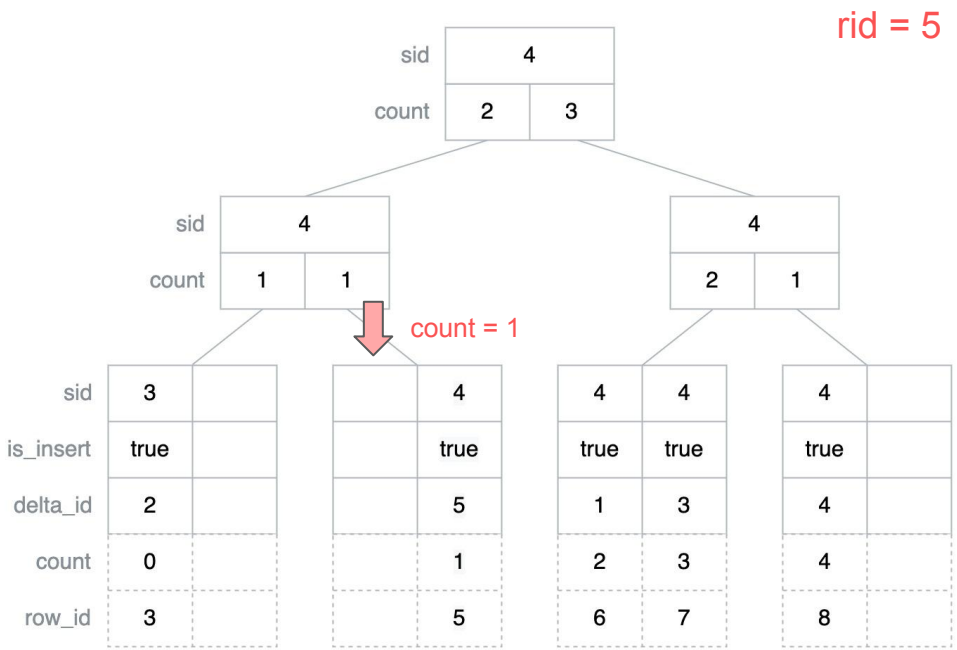

```
leaf, count = findRightLeafByRId(rid)
pos, count = searchLeafForRId(leaf, rid, count)
shiftLeafEntries(leaf, pos, 1)
leaf[pos].sid = rid - count
leaf[pos].delta_id = offset_in_delta_value_space
```

Delta Index: Add Insert



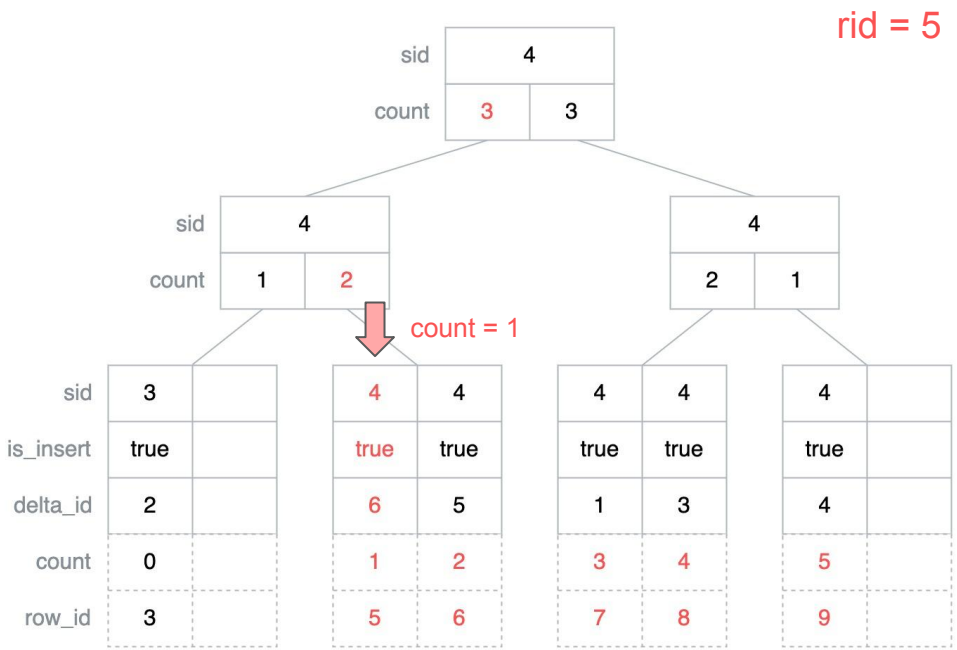

```
leaf, count = findRightLeafByRId(rid)
pos, count = searchLeafForRId(leaf, rid, count)
shiftLeafEntries(leaf, pos, 1)
leaf[pos].sid = rid - count
leaf[pos].delta_id = offset_in_delta_value_space
```

Delta Index: Add Insert



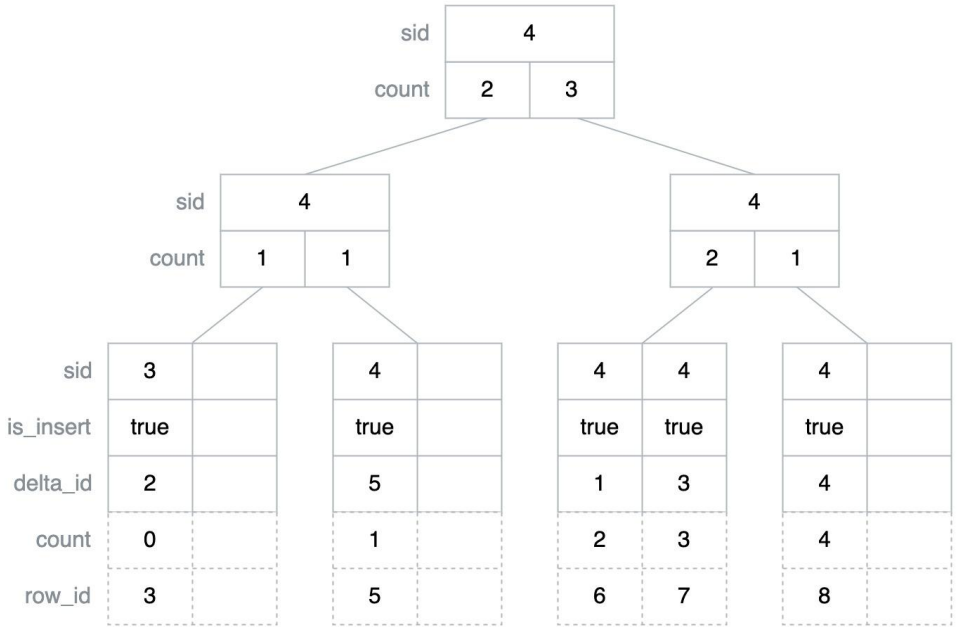

```
leaf, count = findRightLeafByRId(rid)
pos, count = searchLeafForRId(leaf, rid, count)
shiftLeafEntries(leaf, pos, 1)
leaf[pos].sid = rid - count
leaf[pos].delta_id = offset_in_delta_value_space
```

Delta Index: Add Insert



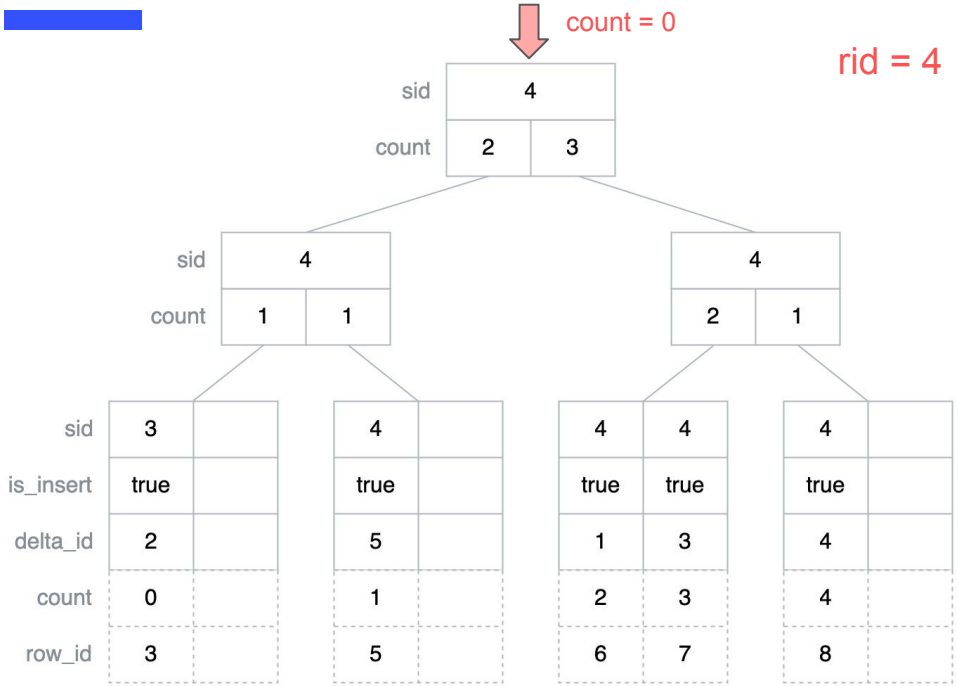

```
leaf, count = findRightLeafByRId(rid)
pos, count = searchLeafForRId(leaf, rid, count)
shiftLeafEntries(leaf, pos, 1)
leaf[pos].sid = rid - count
leaf[pos].is_insert = true
leaf[pos].delta_id = offset_in_delta_value_space
```

Delta Index: Add Delete




```
leaf, count = findRightLeafByRId(rid)
pos, count = searchLeafForRId(leaf, rid, count)
// skip delete chain
while leaf[pos].sid + count == rid {
    if leaf[pos].is_insert {
        break
    }
    pos += 1
    count -= 1
}
if leaf[pos].sid + count == rid {
    shiftLeafEntries(leaf, pos + 1, -1)
} else {
    shiftLeafEntries(leaf, pos, 1)
    leaf[pos].sid = rid - count
    leaf[pos].is_insert = false
}
```

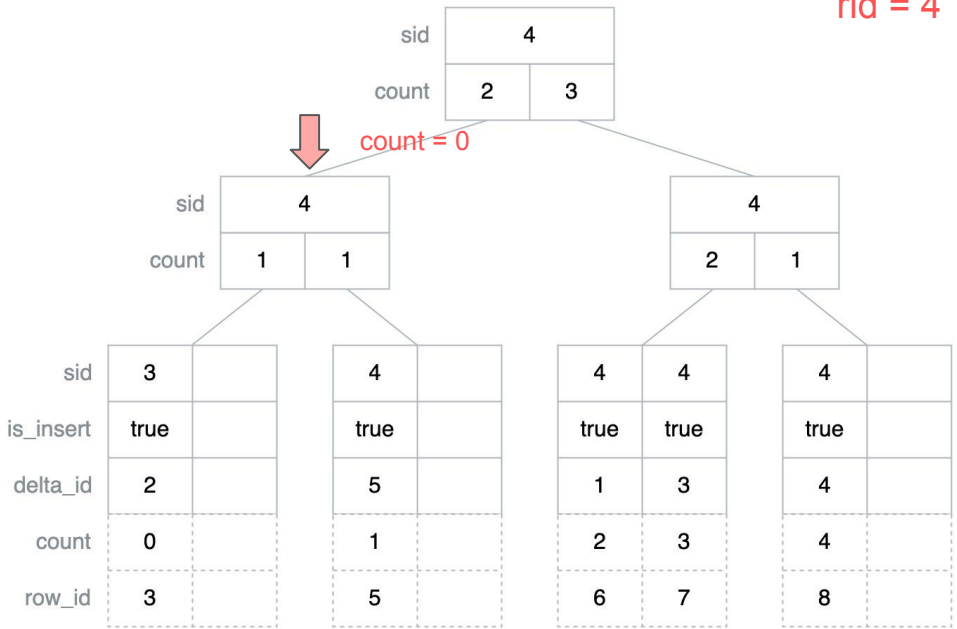
Delta Index: Add Delete



rid = 4

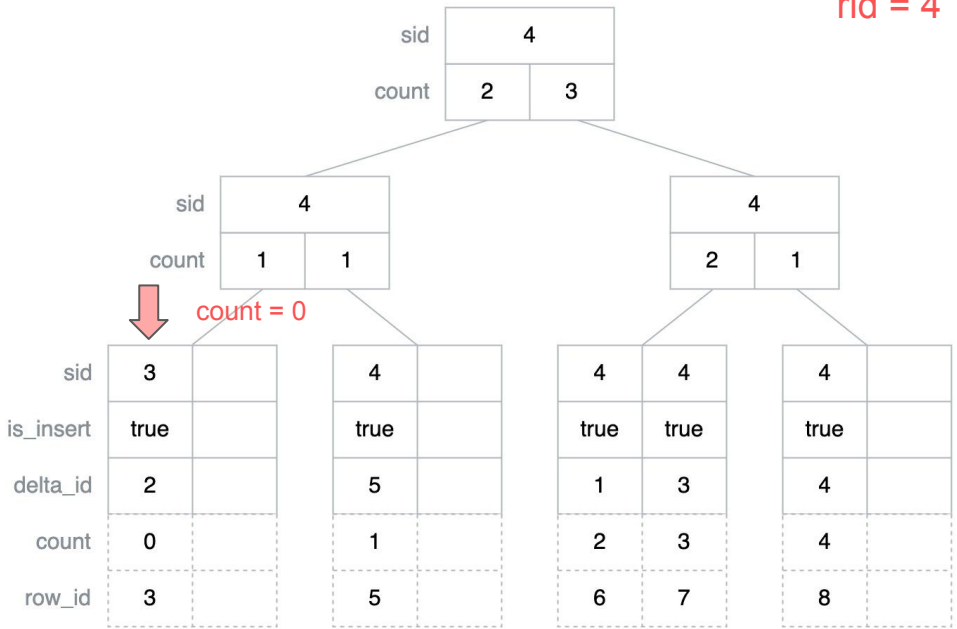
```
leaf, count = findRightLeafByRId(rid)
pos, count = searchLeafForRId(leaf, rid, count)
// skip delete chain
while leaf[pos].sid + count == rid {
    if leaf[pos].is_insert {
        break
    }
    pos += 1
    count -= 1
}
if leaf[pos].sid + count == rid {
    shiftLeafEntries(leaf, pos + 1, -1)
} else {
    shiftLeafEntries(leaf, pos, 1)
    leaf[pos].sid = rid - count
    leaf[pos].is_insert = false
}
```

Delta Index: Add Delete



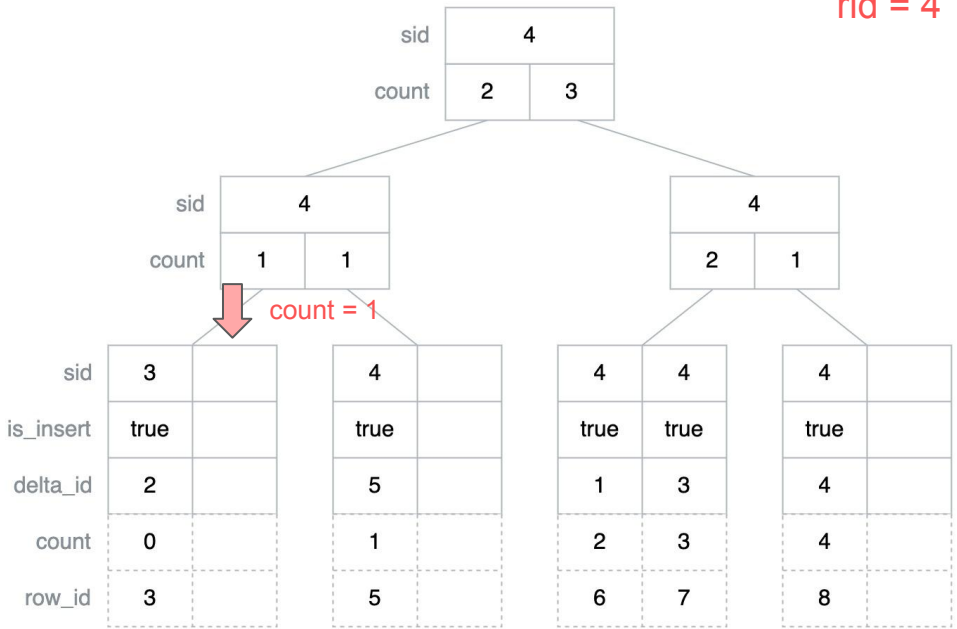

```
leaf, count = findRightLeafByRId(rid)
pos, count = searchLeafForRId(leaf, rid, count)
// skip delete chain
while leaf[pos].sid + count == rid {
    if leaf[pos].is_insert {
        break
    }
    pos += 1
    count -= 1
}
if leaf[pos].sid + count == rid {
    shiftLeafEntries(leaf, pos + 1, -1)
} else {
    shiftLeafEntries(leaf, pos, 1)
    leaf[pos].sid = rid - count
    leaf[pos].is_insert = false
}
```

Delta Index: Add Delete



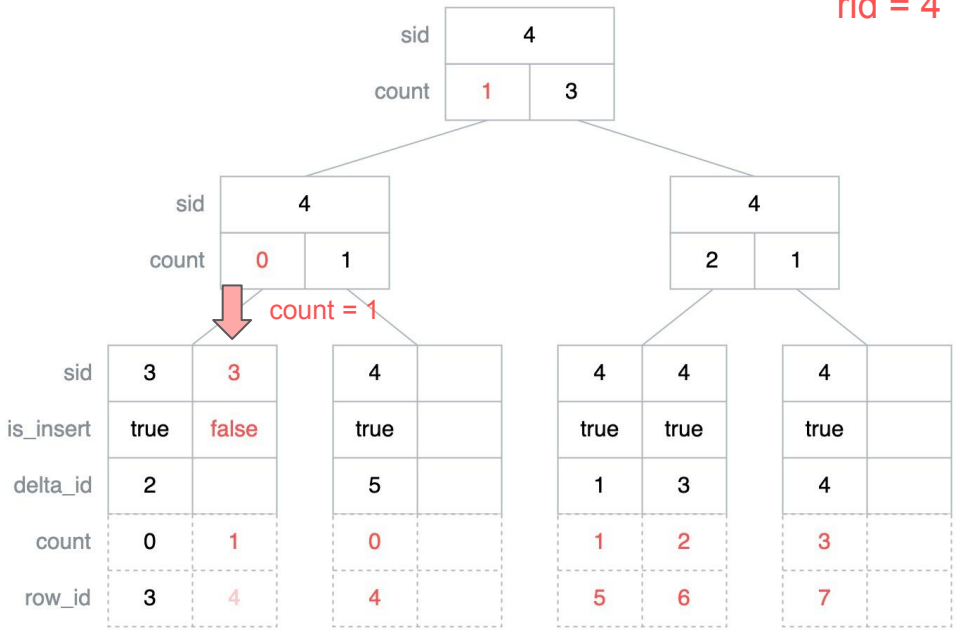

```
leaf, count = findRightLeafByRId(rid)
pos, count = searchLeafForRId(leaf, rid, count)
// skip delete chain
while leaf[pos].sid + count == rid {
    if leaf[pos].is_insert {
        break
    }
    pos += 1
    count -= 1
}
if leaf[pos].sid + count == rid {
    shiftLeafEntries(leaf, pos + 1, -1)
} else {
    shiftLeafEntries(leaf, pos, 1)
    leaf[pos].sid = rid - count
    leaf[pos].is_insert = false
}
```

Delta Index: Add Delete



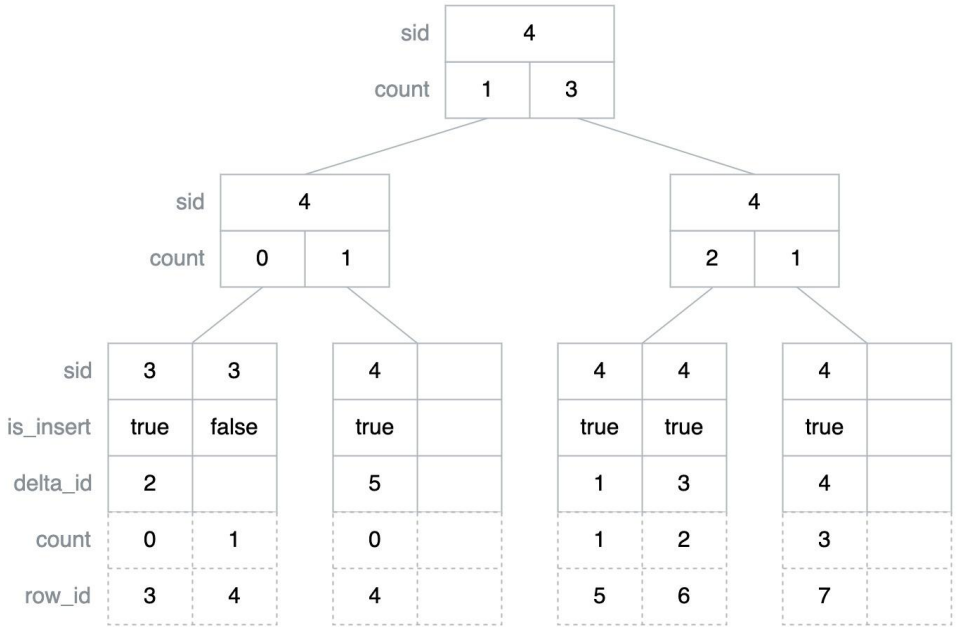

```
leaf, count = findRightLeafByRId(rid)
pos, count = searchLeafForRId(leaf, rid, count)
// skip delete chain
while leaf[pos].sid + count == rid {
    if leaf[pos].is_insert {
        break
    }
    pos += 1
    count -= 1
}
if leaf[pos].sid + count == rid {
    shiftLeafEntries(leaf, pos + 1, -1)
} else {
    shiftLeafEntries(leaf, pos, 1)
    leaf[pos].sid = rid - count
    leaf[pos].is_insert = false
}
```

Delta Index: Add Delete



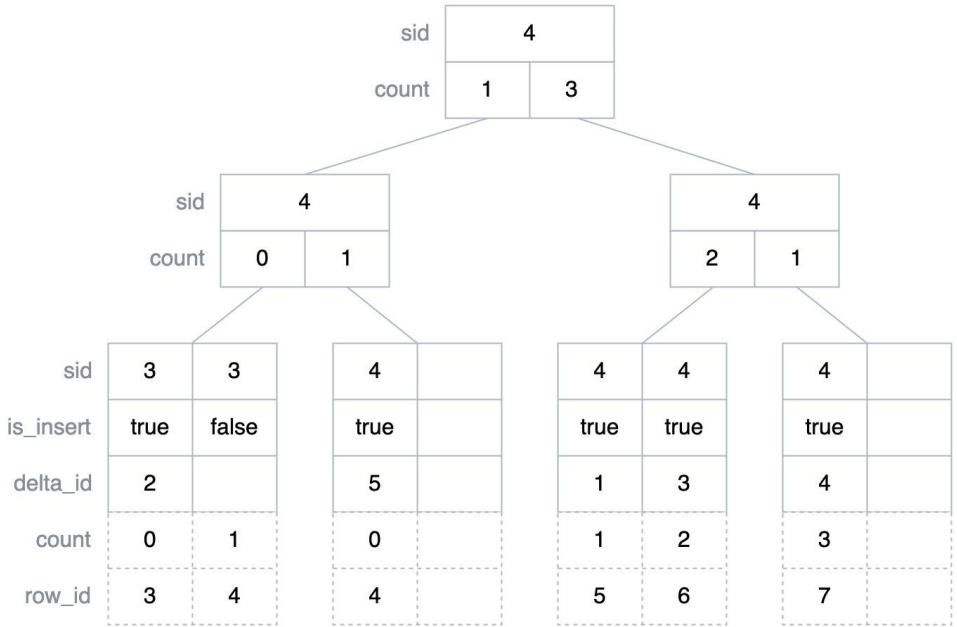

```
leaf, count = findRightLeafByRId(rid)
pos, count = searchLeafForRId(leaf, rid, count)
// skip delete chain
while leaf[pos].sid + count == rid {
    if leaf[pos].is_insert {
        break
    }
    pos += 1
    count -= 1
}
if leaf[pos].sid + count == rid {
    shiftLeafEntries(leaf, pos + 1, -1)
} else {
    shiftLeafEntries(leaf, pos, 1)
    leaf[pos].sid = rid - count
    leaf[pos].is_insert = false
}
```

Delta Index: Read with Delete Entry




```
total_stable_rows = 0
iter = index.begin()
while iter != index.end() {
    if iter->is_insert {
        rows = iter->sid - total_stable_rows
        read_stable_rows(rows)
        read_delta_row(delta_id)
        total_stable_rows += stable_rows
    } else {
        ignore_stable_rows(1)
        total_stable_rows += 1
    }
    iter++
}
```

Delta Index: Add Insert with Delete Entry




```
leaf, count = findRightLeafByRId(rid)
pos, count = searchLeafForRId(leaf, rid, count)
// skip delete chain
while leaf[pos].sid + count == rid {
    if leaf[pos].is_insert {
        break
    }
    pos += 1
    count -= 1
}
shiftLeafEntries(leaf, pos, 1)
leaf[pos].sid = rid - count
leaf[pos].delta_id = offset_in_delta_value_space
```

Minmax Index

DTFile

Handle ↑	Version ↑	ColA
Albedo	**1**	28
Kazuha	**5**	10
Kazuha	**6**	26

Handle ↑	Version ↑	ColA
Kazuha	**7**	37
Klee	**1**	63
Klee	**2**	92
Miko	**1**	62
Miko	**7**	53

SELECT … WHERE ColA < 30

Minmax Index

DTFile

Handle ↑	Version ↑	ColA
Albedo	**1**	28
Kazuha	**5**	10
Kazuha	**6**	26

Handle ↑	Version ↑	ColA
~~**Kazuha**~~	~~**7**~~	~~37~~
~~**Klee**~~	~~**4**~~	~~63~~
~~**Klee**~~	~~**2**~~	~~92~~
~~**Miko**~~	~~**4**~~	~~62~~
~~**Miko**~~	~~**7**~~	~~53~~

SELECT … WHERE ColA < 30

Query Timestamp = 7

Minmax Index

DTFile

Handle ↑	Version ↑	ColA
Albedo	**1**	28
Kazuha	**5**	10
Kazuha	**6**	26
Kazuha	**7**	37

Handle ↑	Version ↑	ColA
~~**Klee**~~	~~**1**~~	~~63~~
~~**Klee**~~	~~**2**~~	~~92~~
~~**Miko**~~	~~**1**~~	~~62~~
~~**Miko**~~	~~**7**~~	~~53~~

SELECT … WHERE ColA < 30

Query Timestamp = 7

How to Get RowID for New Row?

ColumnFileTiny[0]

Handle ↑	Version ↑	ColA
Klee	**3**	40

ColumnFileTiny[1]

Handle ↑	Version ↑	ColA
Kazuha	**10**	1
Kazuha	**15**	5

Delta

DTFile

Handle ↑	Version ↑	ColA
Albedo	**1**	28
Kazuha	**5**	10
Kazuha	**6**	26
Kazuha	**7**	37

Handle ↑	Version ↑	ColA
Klee	**1**	23
Klee	**2**	92
Miko	**1**	62
Miko	**7**	13

Stable

Handle ↑	Version ↑	ColA
Albedo	**1**	28
Kazuha	**5**	10
Kazuha	**6**	26
Kazuha	**7**	37
Kazuha	**10**	1
Kazuha	**15**	5
…		
Miko	**1**	62
Miko	**7**	13

Merged Stream

Handle ↑	Version ↑	ColA
Miko	**4**	60

RowID?

Batch Place

ColumnFileTiny[0]

Handle ↑	Version ↑	ColA
Klee	**3**	40

ColumnFileTiny[1]

Handle ↑	Version ↑	ColA
Kazuha	**10**	1
Kazuha	**15**	5

Delta

DTFile

Handle ↑	Version ↑	ColA
~~**Albedo**~~	~~**4**~~	~~28~~
~~**Kazuha**~~	~~**5**~~	~~10~~
~~**Kazuha**~~	~~**6**~~	~~26~~
~~**Kazuha**~~	~~**7**~~	~~37~~

Handle ↑	Version ↑	ColA
Klee	**1**	23
Klee	**2**	92
Miko	**1**	62
Miko	**7**	13

Stable

Handle ↑	Version ↑	ColA
Kazuha	**10**	1
Kazuha	**15**	5
Klee	**1**	23
Klee	**2**	92
Klee	**3**	40
Miko	**1**	62
Miko	**7**	13

Merged Stream

Handle ↑	Version ↑	ColA
Klee	**8**	40
Miko	**4**	60
Miko	**8**	60

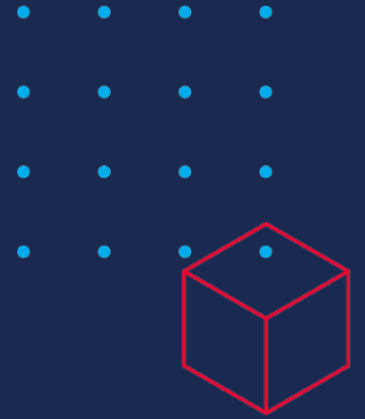
Source Code

DeltaTree
Storages/DeltaMerge/DeltaTree.h
Storages/DeltaMerge/DeltaPlace.h

DeltaMergeBlockInputStream
Storages/DeltaMerge/DeltaMerge.h

Place Rows And Deletes
Storages/DeltaMerge/Segment.cpp#L1469

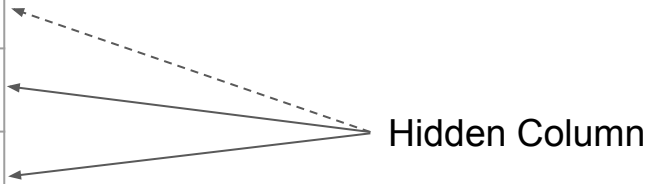
PingCAP TiDB



Example

Example: Schema

Column Name	Column Type
Handle	UInt64
Version	UInt64
DelMark	UInt8
ColA	UInt64



Example: Stable

Handle	Version	DelMark	ColA
1	1	0	11
2	2	0	18
3	3	0	58
4	4	0	62

Example: Insert Rows

sid	4	4
is_insert	true	true
delta_id	0	1

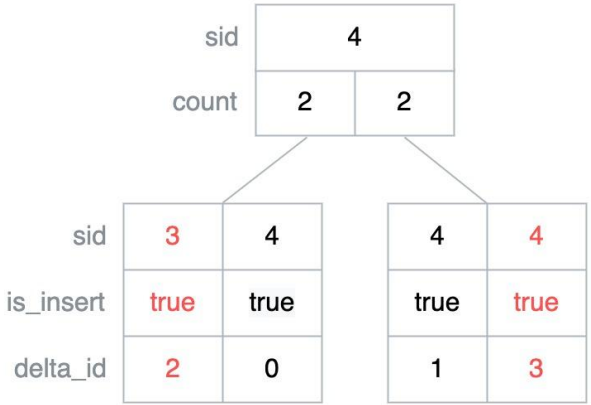
Delta

	Handle	Version	DelMark	ColA
0	5	15	0	46
1	100	16	0	88

Stable

Handle	Version	DelMark	ColA
1	11	0	11
2	12	0	18
3	13	0	58
4	14	0	62

Example: Update Rows



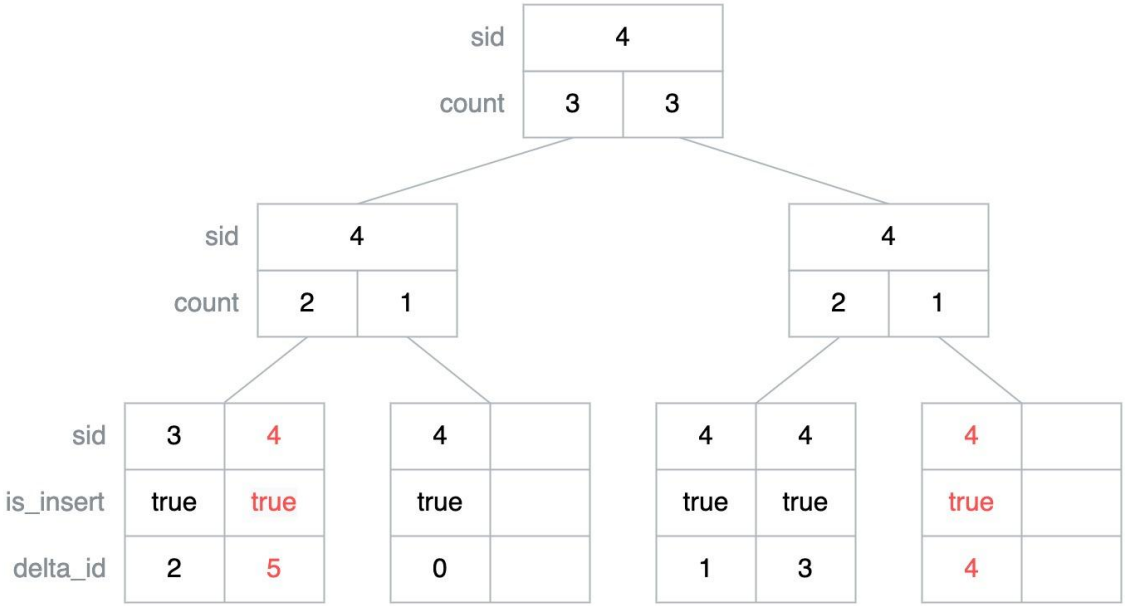
Delta

	Handle	Version	DelMark	ColA
0	5	15	0	46
1	100	16	0	88
2	3	17	0	61
3	100	17	0	28

Stable

Handle	Version	DelMark	ColA
1	11	0	11
2	12	0	18
3	13	0	58
4	14	0	62

Example: Delete Rows



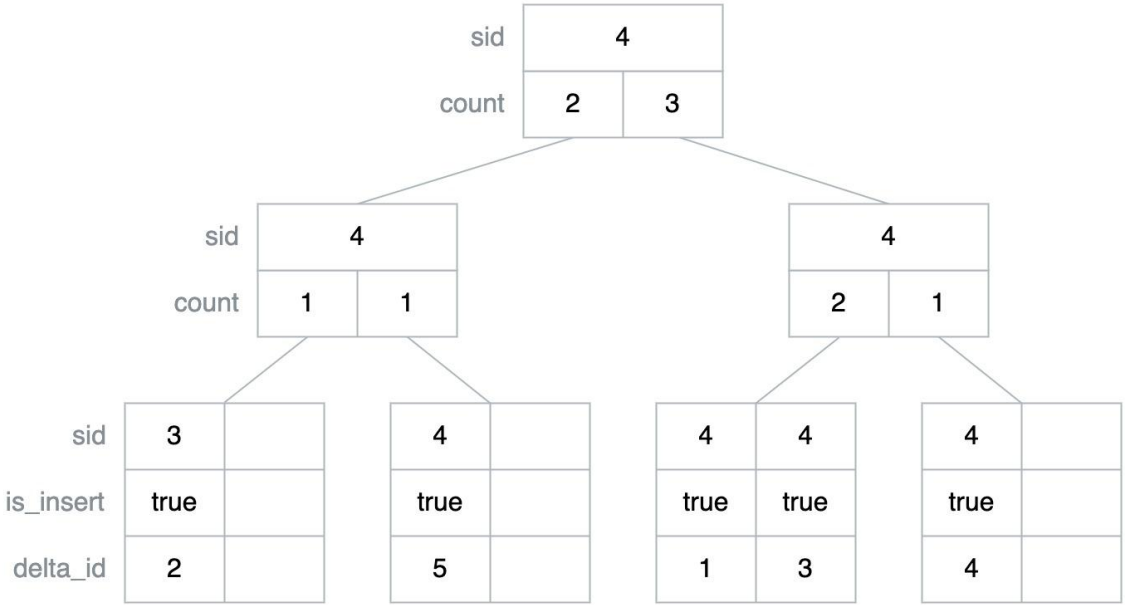
Delta

	Handle	Version	DelMark	ColA
0	5	15	0	46
1	100	16	0	88
2	3	17	0	61
3	100	17	0	28
4	100	18	1	
5	4	18	1	

Stable

Handle	Version	DelMark	ColA
1	11	0	11
2	12	0	18
3	13	0	58
4	14	0	62

Example: Delete Range [5, 100)



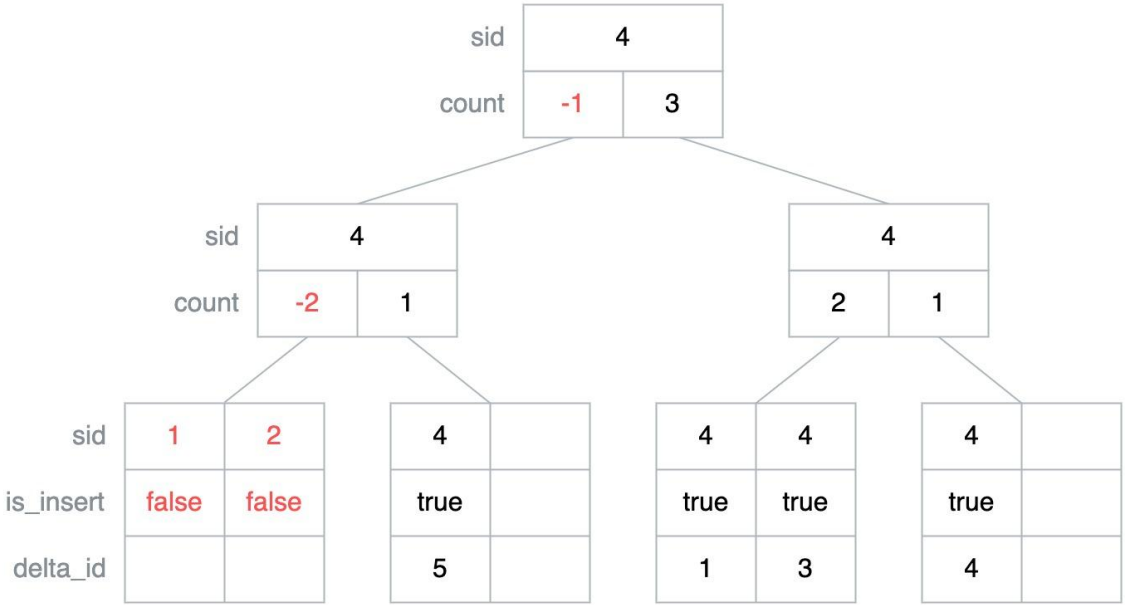
Delta

	Handle	Version	DelMark	ColA
0	~~5~~	~~15~~	~~0~~	~~46~~
1	100	16	0	88
2	3	17	0	61
3	100	17	0	28
4	100	18	1	
5	4	18	1	

Stable

Handle	Version	DelMark	ColA
1	11	0	11
2	12	0	18
3	13	0	58
4	14	0	62

Example: Delete Range [2, 4)



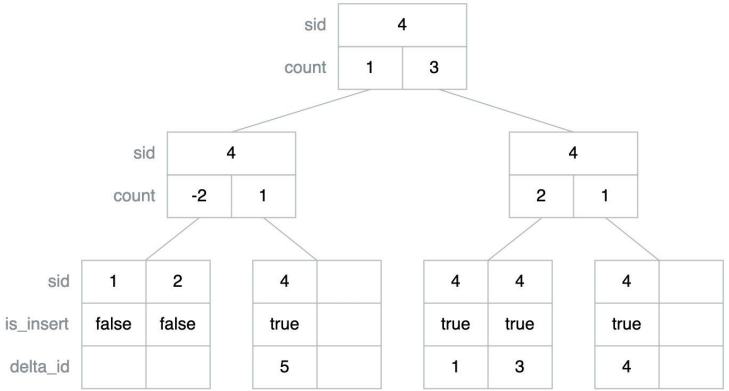
Delta

	Handle	Version	DelMark	ColA
0	~~5~~	~~15~~	~~0~~	~~46~~
1	100	16	0	88
2	~~3~~	~~17~~	~~0~~	~~61~~
3	100	17	0	28
4	100	18	1	
5	4	18	1	

Stable

Handle	Version	DelMark	ColA
1	11	0	11
~~2~~	~~12~~	~~0~~	~~18~~
~~3~~	~~13~~	~~0~~	~~58~~
4	14	0	62

Example: Read



Compact

sid	1	2	4	4	4
is_insert	false	false	true	true	true
delta_id			5	1	3
count			1	1	2

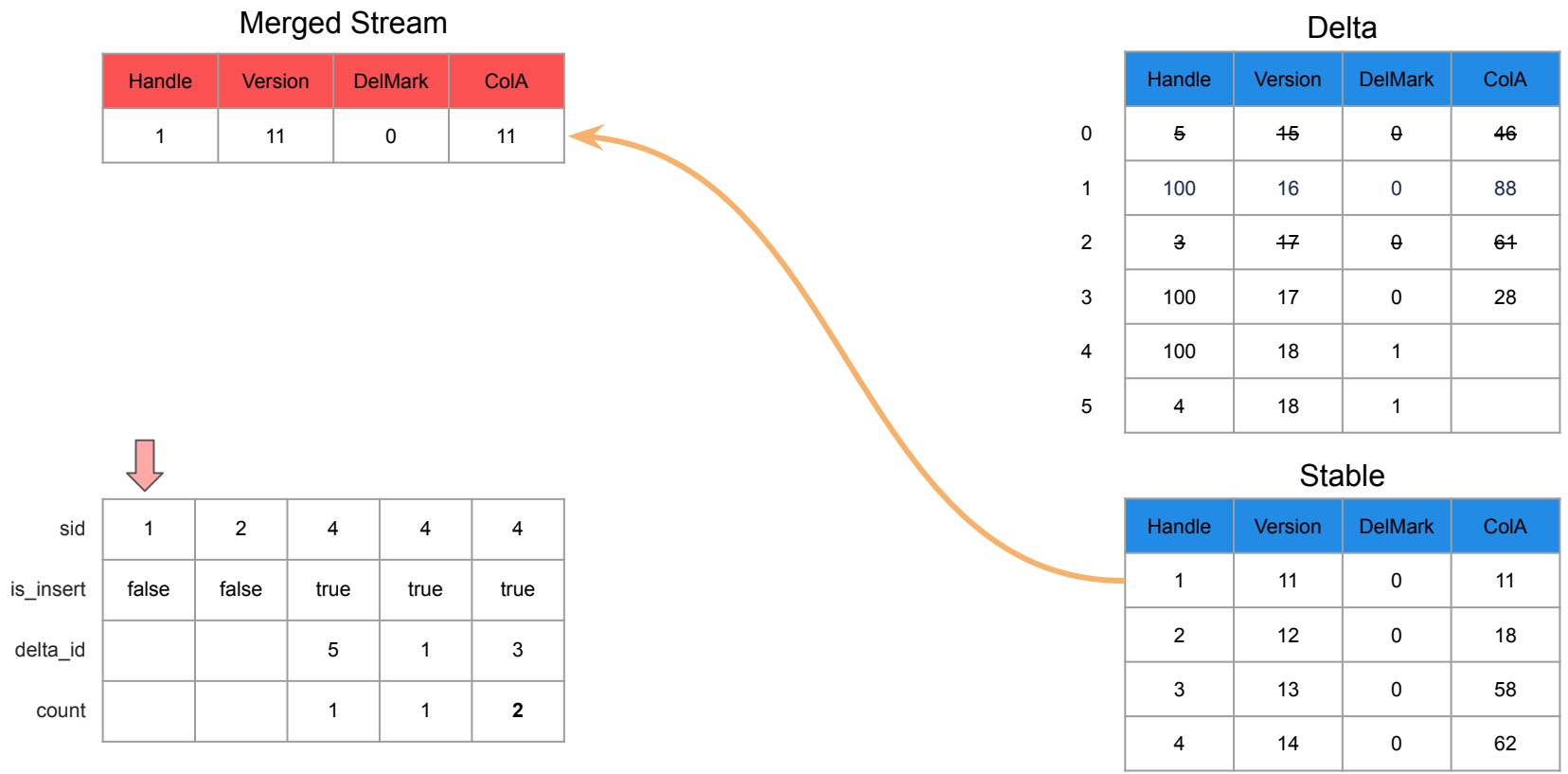
Delta

	Handle	Version	DelMark	ColA
0	~~5~~	~~15~~	~~0~~	~~46~~
1	100	16	0	88
2	~~3~~	~~17~~	~~0~~	~~61~~
3	100	17	0	28
4	100	18	1	
5	4	18	1	

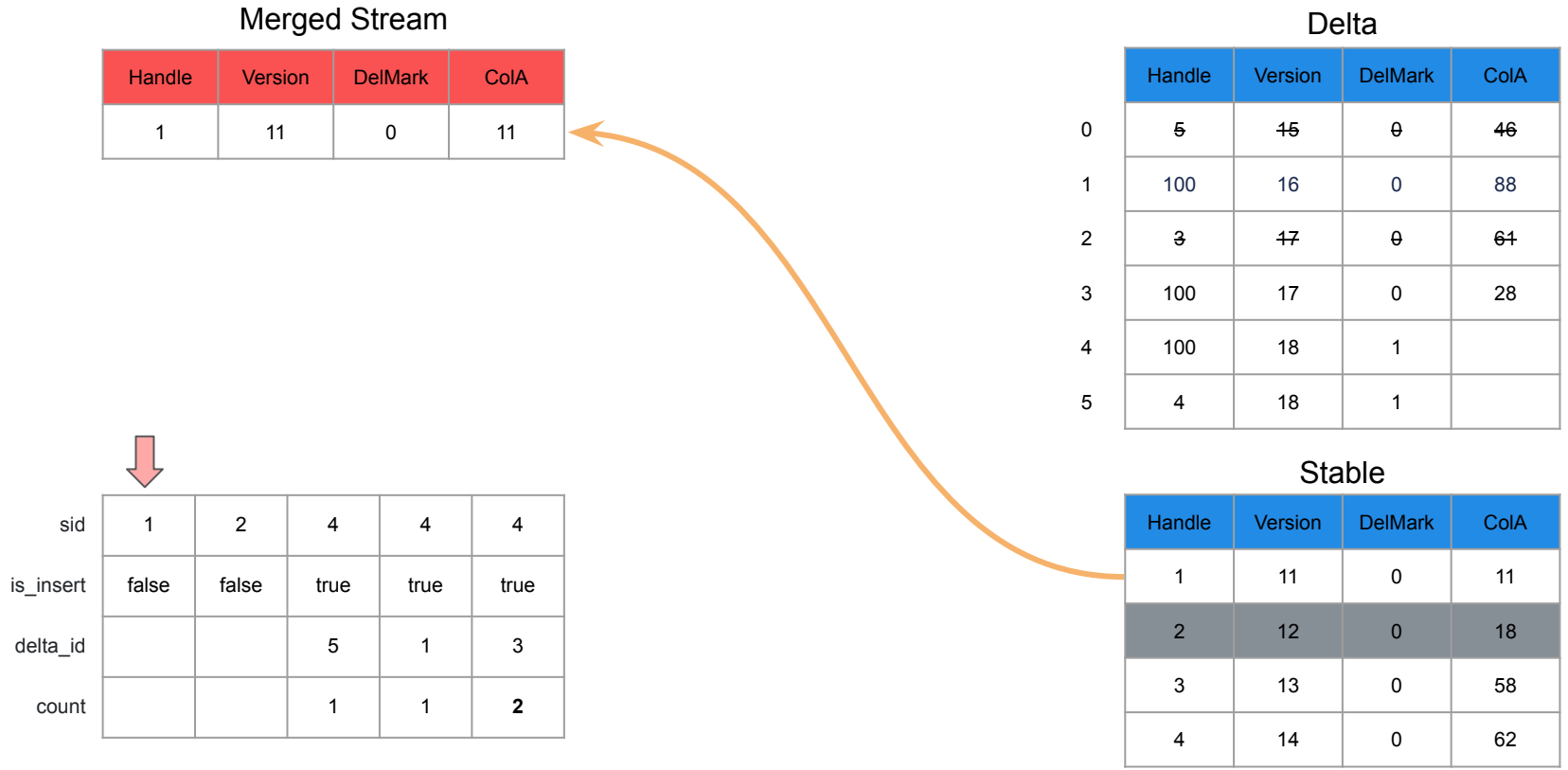
Stable

Handle	Version	DelMark	ColA
1	11	0	11
~~2~~	~~12~~	~~0~~	~~18~~
~~3~~	~~13~~	~~0~~	~~58~~
4	14	0	62

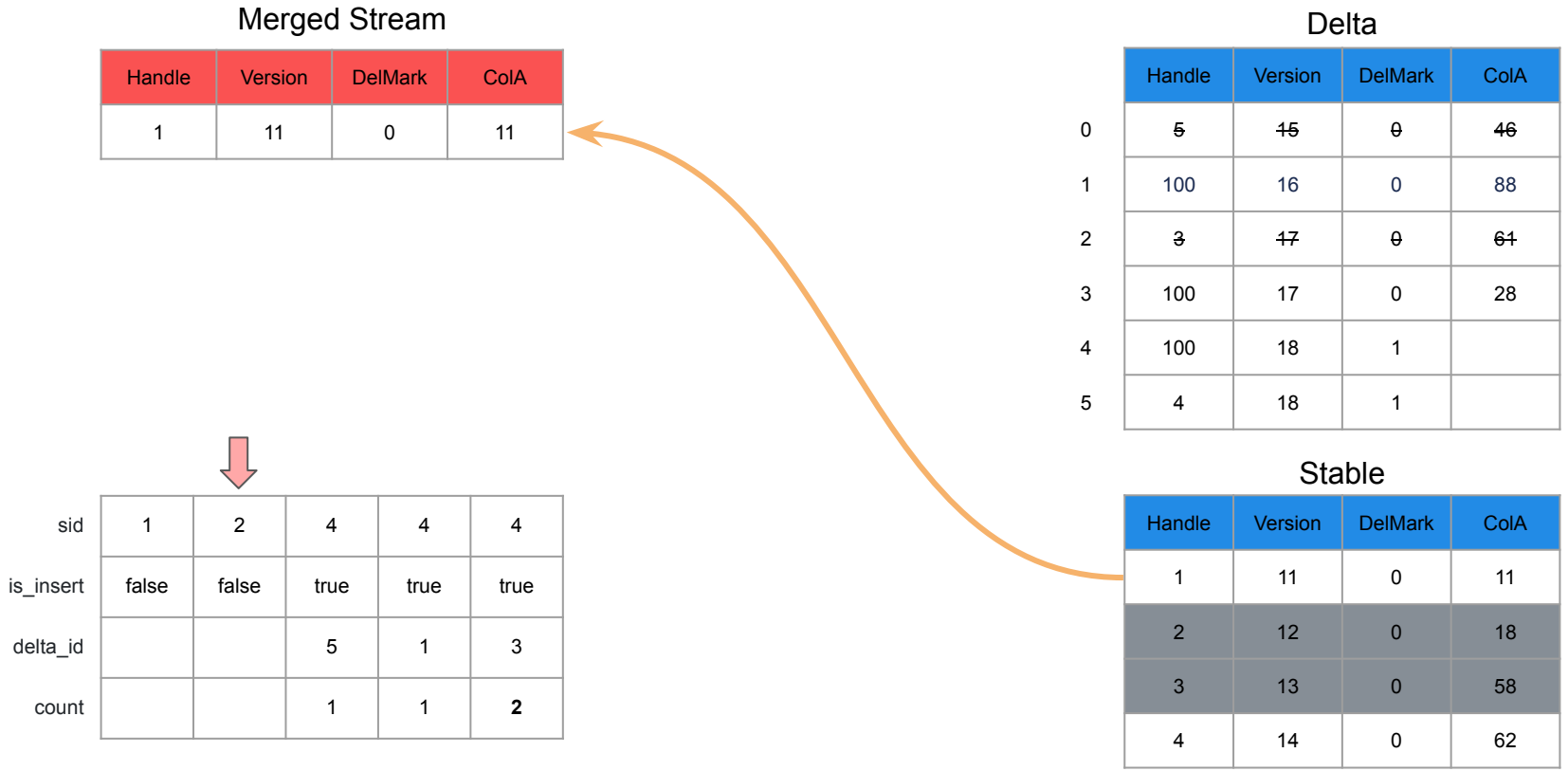
Example: Read



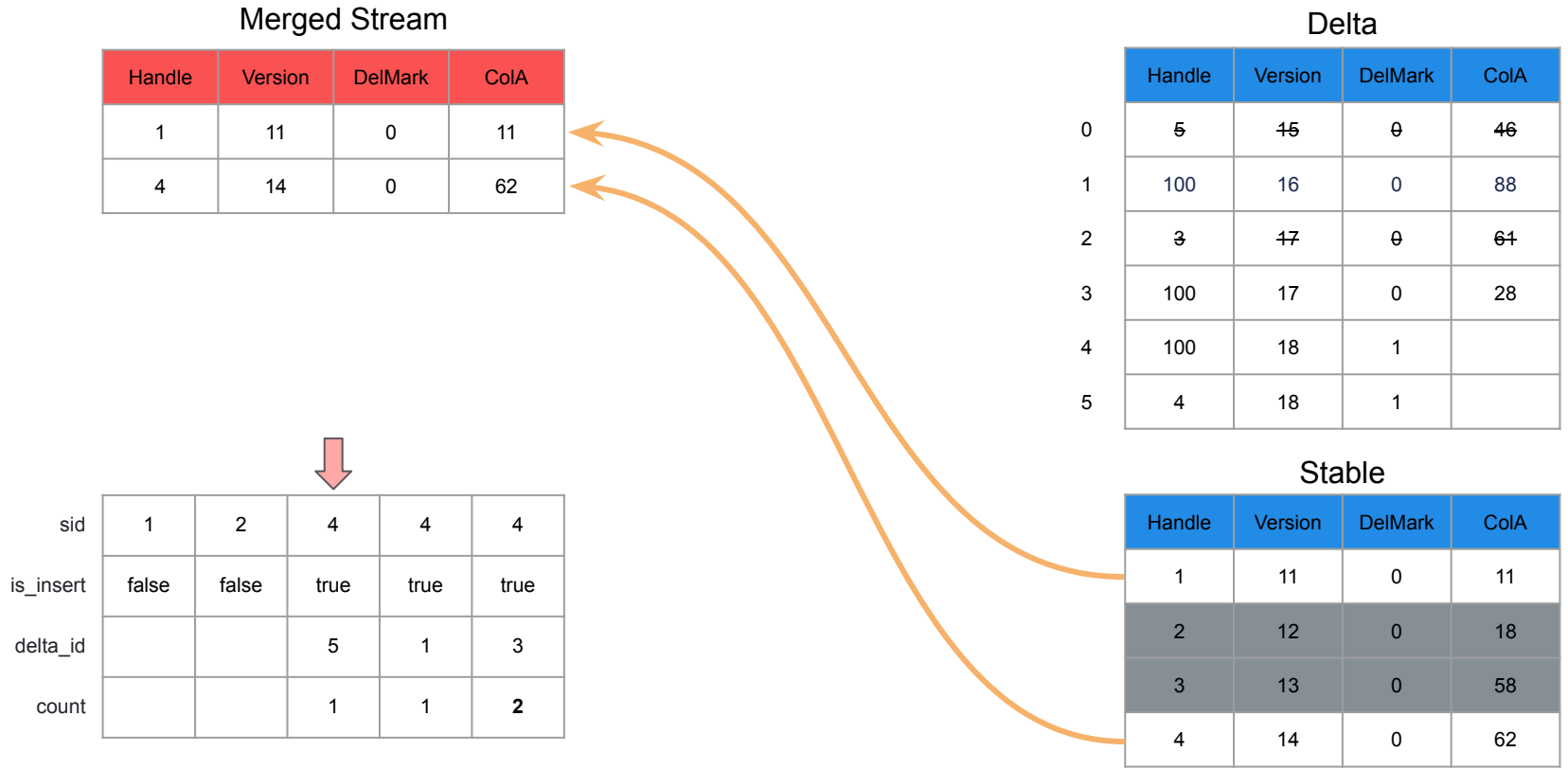
Example: Read



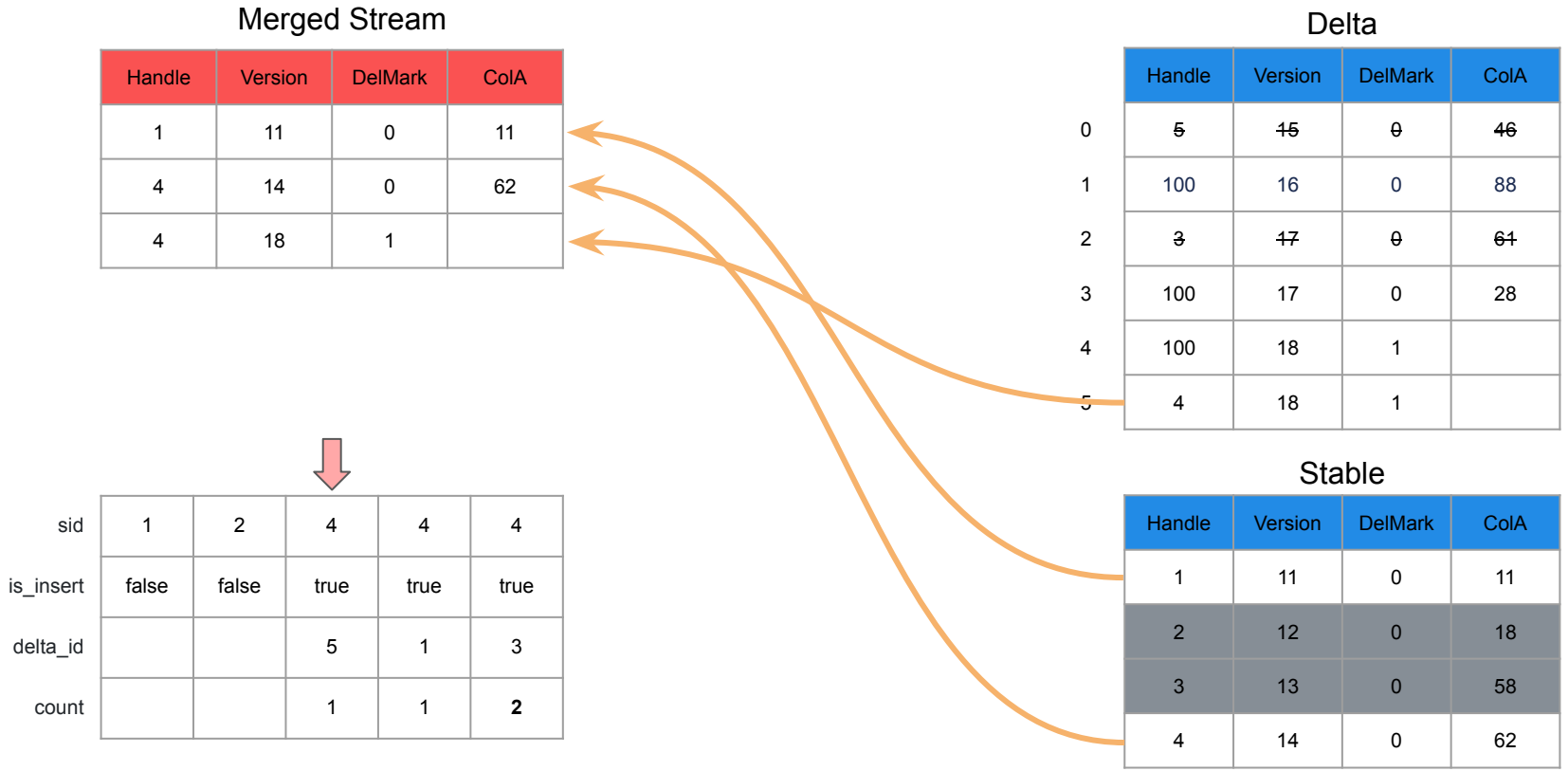
Example: Read



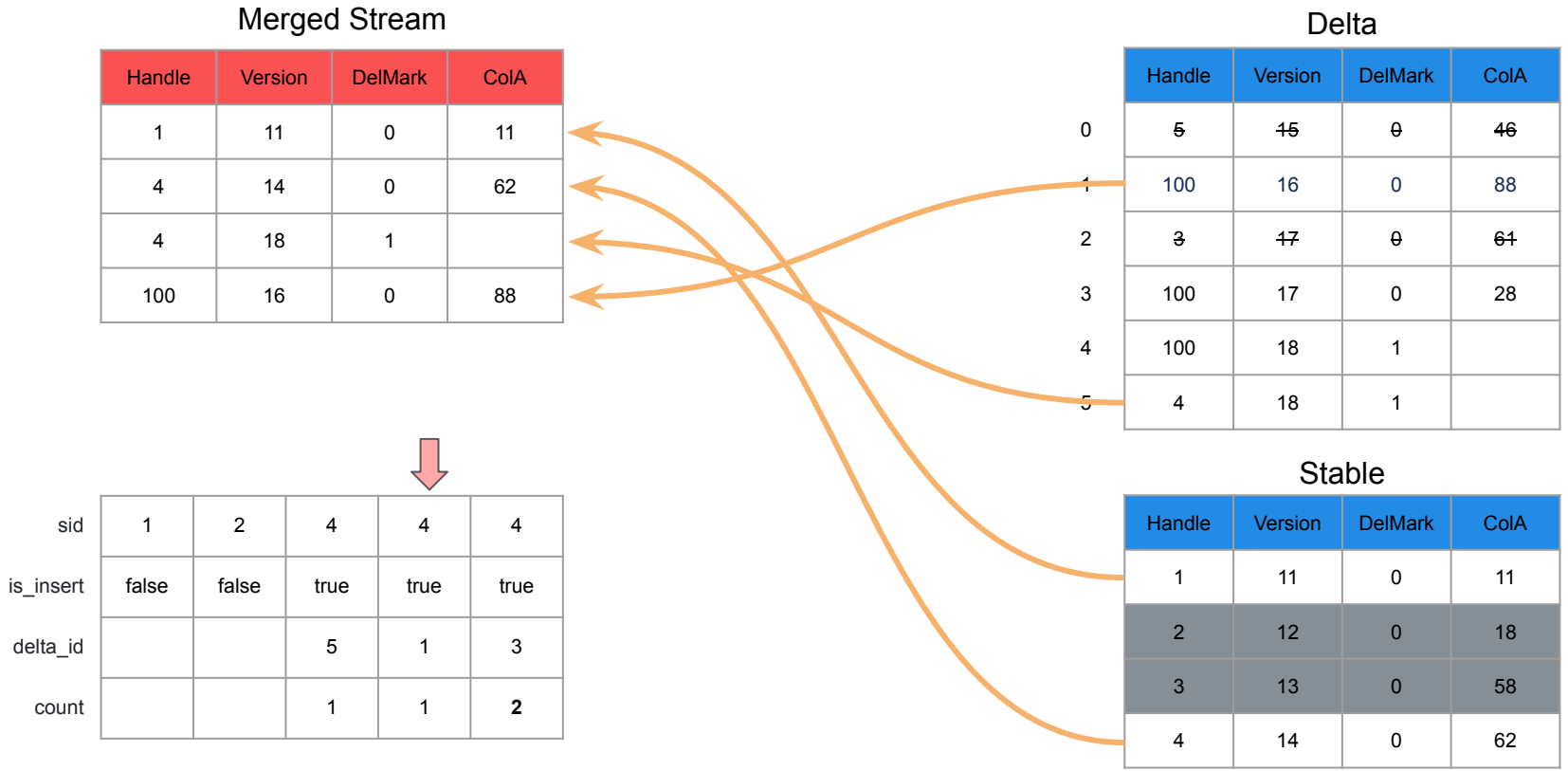
Example: Read



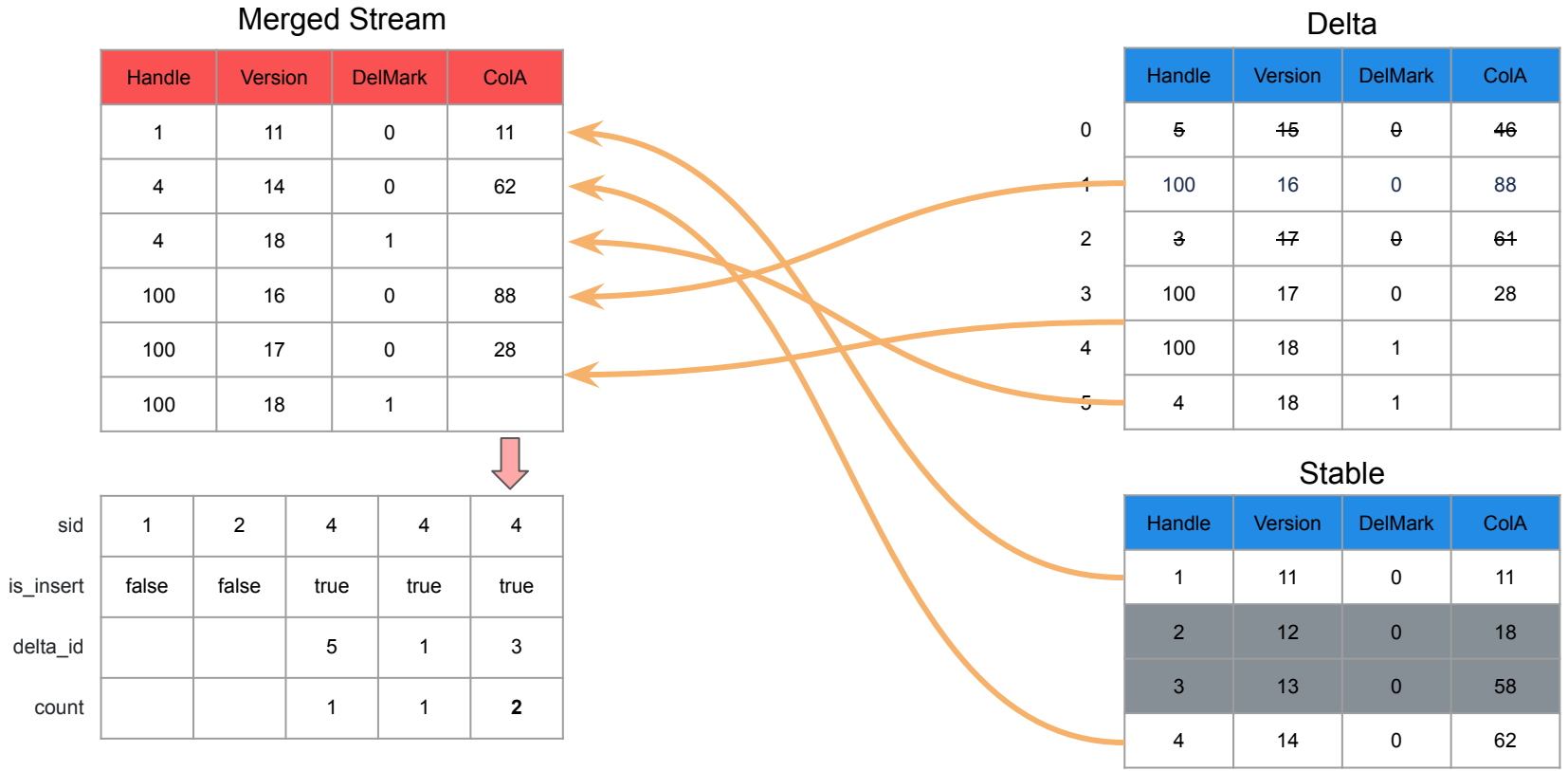
Example: Read



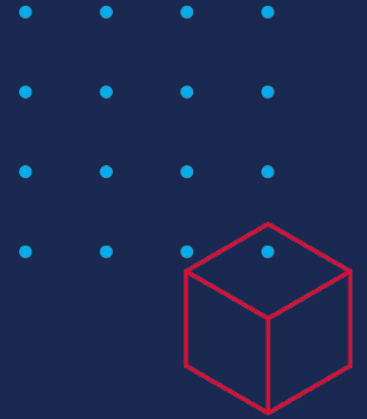
Example: Read



Example: Read

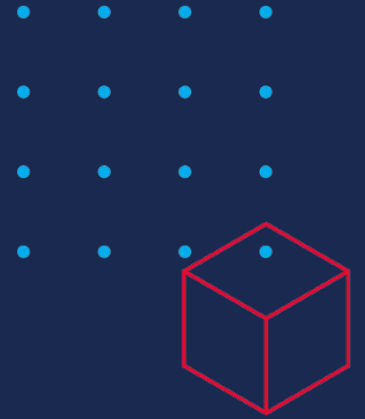


PingCAP TiDB



Q&A

PingCAP TiDB



Thanks