```
                trx_state: RUNNING
              trx_started: 2014-11-19 13:24:40
    trx_requested_lock_id: NULL
         trx_wait_started: NULL
               trx_weight: 586739
      trx_mysql_thread_id: 2
                trx_query: DELETE FROM employees.salaries WHERE salary > 65000
      trx_operation_state: updating or deleting
        trx_tables_in_use: 1
        trx_tables_locked: 1
         trx_lock_structs: 3003
    trx_lock_memory_bytes: 450768
          trx_rows_locked: 1407513
        trx_rows_modified: 583736
   trx_concurrency_tickets: 0
       trx_isolation_level: REPEATABLE READ
         trx_unique_checks: 1
    trx_foreign_key_checks: 1
trx_last_foreign_key_error: NULL
 trx_adaptive_hash_latched: 0
 trx_adaptive_hash_timeout: 10000
          trx_is_read_only: 0
trx_autocommit_non_locking: 0
       trx_schedule_weight: NULL
```

Notes

- Use this table to help diagnose performance problems that occur during times of heavy concurrent load. Its contents are updated as described in Section 15.15.2.3, “Persistence and Consistency of InnoDB Transaction and Locking Information”.
- You must have the `PROCESS` privilege to query this table.
- Use the `INFORMATION_SCHEMA COLUMNS` table or the `SHOW COLUMNS` statement to view additional information about the columns of this table, including data types and default values.

26.4.31 The INFORMATION_SCHEMA INNODB_VIRTUAL Table

The `INNODB_VIRTUAL` table provides metadata about `InnoDB` virtual generated columns and columns upon which virtual generated columns are based.

A row appears in the `INNODB_VIRTUAL` table for each column upon which a virtual generated column is based.

The `INNODB_VIRTUAL` table has these columns:

- `TABLE_ID`

  An identifier representing the table associated with the virtual column; the same value as `INNODB_TABLES.TABLE_ID`.

- `POS`

  The position value of the virtual generated column. The value is large because it encodes the column sequence number and ordinal position. The formula used to calculate the value uses a bitwise operation:

  ```
  ((nth virtual generated column for the InnoDB instance + 1) << 16)
  + the ordinal position of the virtual generated column
  ```

  For example, if the first virtual generated column in the `InnoDB` instance is the third column of the table, the formula is `(0 + 1) << 16) + 2`. The first virtual generated column in the `InnoDB` instance is