PERCONA

Survey of Percona Toolkit:
Command-line Tools for MySQL

Bill Karwin, Percona Inc.

Percona Toolkit

- Free, open source tools based on Percona's experience developing best practices for repetitive or complex MySQL operations.
  - Verify master and replica data consistency.
  - Efficiently archive rows.
  - Find duplicate indexes.
  - Summarize MySQL servers.
  - Analyze queries from logs and tcpdump.
  - Collect diagnostic information when problems occur.

Installing

Installing Percona Toolkit

- Percona package repos for RHEL/CentOS or Debian/Ubuntu:

```
$ sudo yum install percona-toolkit
$ sudo dpkg install percona-toolkit
```

  - http://www.percona.com/software/repositories

- Other download options available for RPM, DEB, tarball, or individual tools.
  - http://www.percona.com/doc/percona-toolkit/installation.html

Top Nine Popular Tools

- pt-summary
- pt-mysql-summary
- pt-stalk
- pt-archiver
- pt-query-digest
- pt-duplicate-key-checker
- pt-table-checksum
- pt-table-sync
- pt-online-schema-change

pt-summary

http://www.percona.com/doc/percona-toolkit/pt-summary.html

pt-summary

- Summarize system information in a nice way.
- Useful to verify operating system configuration, inspect many system attributes quickly.

pt-summary

```
$ pt-summary
# Percona Toolkit System Summary Report ######################
        Date | 2011-09-30 17:06:44 UTC (local TZ: PDT -0700)
    Hostname | huey.karwin.percona.com
      Uptime | 7:45, 1 user, load average: 0.04, 0.01, 0.00
      System | innotek GmbH; VirtualBox; v1.2 ()
 Service Tag | 0
    Platform | Linux
     Release | CentOS release 5.6 (Final)
      Kernel | 2.6.18-238.19.1.el5
Architecture | CPU = 64-bit, OS = 64-bit
   Threading | NPTL 2.5
    Compiler | GNU CC version 4.1.2 20080704 (Red Hat 4.1.2-51).
     SELinux | Enforcing
 Virtualized | VirtualBox
# Processor ##################################################
  Processors | physical = 1, cores = 0, virtual = 1, hyperthreading = no
      Speeds | 1x2844.667
      Models | 1xIntel(R) Core(TM) i7 CPU M 640 @ 2.80GHz
      Caches | 1x6144 KB
```

pt-summary

```
# Memory #####################################################
       Total | 497.86M
        Free | 44.36M
        Used | physical = 453.50M, swap = 0.00k, virtual = 453.50M
     Buffers | 53.18M
      Caches | 260.91M
       Dirty | 48 kB
     UsedRSS | 113.4M
  Swappiness | vm.swappiness = 60
 DirtyPolicy | vm.dirty_ratio = 40, vm.dirty_background_ratio = 10
 DirtyStatus | vm.dirty_bytes = 0, vm.dirty_background_bytes = 0
```

pt-summary

```
# Mounted Filesystems ########################################
  Filesystem                       Size Used Type  Opts Mountpoint
  /dev/mapper/VolGroup00-LogVol00   15G  14% ext3  rw   /
  /dev/sda1                         99M  21% ext3  rw   /boot
  tmpfs                            249M   0% tmpfs rw   /dev/shm
# Disk Schedulers And Queue Size #############################
         hdc | [cfq] 128
         sda | [cfq] 128
# Disk Partioning ############################################
Device       Type      Start        End               Size
============ ==== ========== ========== ==================
/dev/sda     Disk                              17179869184
/dev/sda1    Part          1         13           98703360
/dev/sda2    Part         14       2088        17059230720
# Kernel Inode State #########################################
dentry-state | 35813 33772 45 0 0 0
     file-nr | 510 0 49646
    inode-nr | 29137 75
# LVM Volumes ################################################
  LV       VG         Attr   LSize  Origin Snap%  Move Log Copy%  Convert
  LogVol00 VolGroup00 -wi-ao 14.88G
  LogVol01 VolGroup00 -wi-ao  1.00G
```

pt-summary

```
# Network Config #############################################
  Controller | Intel Corporation 82540EM Gigabit Ethernet Controller (rev 02)
  Controller | Intel Corporation 82540EM Gigabit Ethernet Controller (rev 02)
 FIN Timeout | net.ipv4.tcp_fin_timeout = 60
  Port Range | net.ipv4.ip_local_port_range = 32768 61000
# Interface Statistics #######################################
  interface  rx_bytes rx_packets  rx_errors   tx_bytes tx_packets  tx_errors
  ========= ========= ========== ========== ========== ========== ==========
  lo             7000         60          0       7000         60          0
  eth0        1250000      17500          0    3000000      15000          0
  eth1        9000000      12500          0     600000       8000          0
# Network Connections ########################################
  Connections from remote IP addresses
    192.168.56.1        1
  Connections to local IP addresses
    192.168.56.111      1
  Connections to top 10 local ports
    22                  1
  States of connections
    ESTABLISHED         1
    LISTEN              5
```

pt-summary

```
# Top Processes ##############################################
  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
    1 root      15   0 10372  688  572 S  0.0  0.1   0:00.44 init
    2 root      RT  -5     0    0    0 S  0.0  0.0   0:00.00 migration/0
    3 root      34  19     0    0    0 S  0.0  0.0   0:00.20 ksoftirqd/0
    4 root      RT  -5     0    0    0 S  0.0  0.0   0:00.03 watchdog/0
    5 root      10  -5     0    0    0 S  0.0  0.0   0:07.58 events/0
    6 root      10  -5     0    0    0 S  0.0  0.0   0:00.00 khelper
   11 root      10  -5     0    0    0 S  0.0  0.0   0:00.00 kthread
   15 root      10  -5     0    0    0 S  0.0  0.0   0:00.11 kblockd/0
   16 root      20  -5     0    0    0 S  0.0  0.0   0:00.00 kacpid
# Simplified and fuzzy rounded vmstat (wait please) ##########
  procs  ---swap-- -----io---- ---system---- --------cpu--------
   r  b    si   so    bi    bo     ir     cs  us  sy  il  wa  st
   2  0     0    0    10     8   1000     30   0   3  97   0   0
   0  0     0    0     0     0   1000     30   0   2  98   0   0
   0  0     0    0     0     0   1000     30   0   4  96   0   0
   0  0     0    0     0     0   1000     30   0   2  98   0   0
   0  0     0    0     0     0   1000     35   0   2  98   0   0
# The End ####################################################
```

pt-mysql-summary

http://www.percona.com/doc/percona-toolkit/pt-mysql-summary.html

pt-mysql-summary

- Summarize MySQL information in a nice way.
  - See current status at a glance.
  - Uses live information from running instance, because it could be different from `/etc/my.cnf`.
  - Organizes information in a consistent order, so you know where to find it.

pt-mysql-summary

```
$ pt-mysql-summary
# Percona Toolkit MySQL Summary Report #######################
         System time | 2011-09-30 17:57:07 UTC (local TZ: PDT -0700)
# Instances ##################################################
  Port  Data Directory             Socket
  ===== ========================== ======
        /var/lib/mysql
# Report On Port 3306 ########################################
                User | root@localhost
                Time | 2011-09-30 10:57:07 (PDT)
            Hostname | huey.karwin.percona.com
             Version | 5.1.58-community-log MySQL
            Built On | unknown-linux-gnu x86_64
             Started | 2011-09-30 10:25 (up 0+00:31:18)
           Databases | 5
             Datadir | /var/lib/mysql/
           Processes | 1 connected, 1 running
         Replication | Is not a slave, has 0 slaves connected
             Pidfile | /var/lib/mysql/huey.karwin.percona.com.pid (exists)
```

pt-mysql-summary

```
# Processlist ################################################

  Command                        COUNT(*) Working SUM(Time) MAX(Time)
  ------------------------------ -------- ------- --------- ---------
  Binlog Dump                           4       4   1000000    350000
  Query                                 1       1         0         0
  Sleep                                30       0        45         5

  User                           COUNT(*) Working SUM(Time) MAX(Time)
  ------------------------------ -------- ------- --------- ---------
  appuser                              29       0         0         0
  repl                                  1       1     70000     70000

  Host                           COUNT(*) Working SUM(Time) MAX(Time)
  ------------------------------ -------- ------- --------- ---------
  192.168.56.127                       29      11         0         0
  192.168.56.128                        1       1     70000     70000

  db                             COUNT(*) Working SUM(Time) MAX(Time)
  ------------------------------ -------- ------- --------- ---------
  shopsite                             29       0         0         0
  NULL                                  1       1    100000    100000
```

pt-mysql-summary

```
# Status Counters (Wait 10 Seconds) ##########################
Variable                                Per day  Per second     10 secs
Bytes_received                        150000000        1750          90
Bytes_sent                              3000000          35        1500
. . .
Handler_read_rnd_next                     30000                      30
Handler_write                           2250000          25          30
. . .
Queries                                   60000                       2
Questions                                 20000                       2
Select_scan                                 500
Sort_rows                                   175
Sort_scan                                    45
Table_locks_immediate                      4000
Threads_created                             450
Uptime                                    90000           1           1
```

pt-mysql-summary

```
# Table cache ################################################
                Size | 4
               Usage | 100%
# Key Percona Server features ################################
 Table & Index Stats | Not Supported
Multiple I/O Threads | Enabled
Corruption Resilient | Not Supported
 Durable Replication | Not Supported
Import InnoDB Tables | Not Supported
Fast Server Restarts | Not Supported
    Enhanced Logging | Not Supported
Replica Perf Logging | Not Supported
 Response Time Hist. | Not Supported
     Smooth Flushing | Not Supported
 HandlerSocket NoSQL | Not Supported
 Fast Maatkit Hashes | Unknown
# Query cache ################################################
    query_cache_type | ON
                Size | 0.0k
               Usage | 0%
    HitToInsertRatio | 0%
```

pt-mysql-summary

```
# Schema #####################################################
Would you like to mysqldump -d the schema and analyze it? y/n Y
There are 5 databases.  Would you like to dump all, or just one?
Type the name of the database, or press Enter to dump all of them. sakila

  Database Tables Views SPs Trigs Funcs   FKs Partn
  {chosen}     16     7   3           3    22

  Database InnoDB MyISAM
  {chosen}     15      8

  Database BTREE FULLTEXT
  {chosen}    63        1

             s   v   t   t   d   t   y   d   e   s   m   c   i   b
             m   a   i   i   a   e   e   e   n   e   e   h   n   l
             a   r   m   n   t   x   a   c   u   t   d   a   t   o
             l   c   e   y   e   t   r   i   m       i   r       b
             l   h   s   i   t           m           u
             i   a   t   n   i           a           m
             n   r   a   t   m           l           i
             t       m       e                       n
                     p                               t
  Database === === === === === === === === === === === === === ===
  {chosen}  26  45  15  19   4   4   1   7   3   1   2   1   2   1
```

pt-mysql-summary

```
# Noteworthy Technologies ####################################
  Full Text Indexing | Yes
    Geospatial Types | No
        Foreign Keys | Yes
        Partitioning | No
                 SSL | No
Explicit LOCK TABLES | No
      Delayed Insert | No
     XA Transactions | No
         NDB Cluster | No
 Prepared Statements | No
```

pt-mysql-summary

```
# InnoDB #####################################################
             Version | 1.0.17
    Buffer Pool Size | 16.0M
    Buffer Pool Fill | 45%
   Buffer Pool Dirty | 0%
      File Per Table | ON
           Page Size | 16k
       Log File Size | 2 * 5M = 10.0M
     Log Buffer Size | 8M
        Flush Method | O_DIRECT
 Flush Log At Commit | 2
          XA Support | ON
           Checksums | ON
         Doublewrite | ON
     R/W I/O Threads | 4 4
        I/O Capacity | 200
  Thread Concurrency | 0
 Concurrency Tickets | 500
  Commit Concurrency | 0
 Txn Isolation Level | REPEATABLE-READ
   Adaptive Flushing | ON
 Adaptive Checkpoint | 0
      Checkpoint Age | 0k
```

pt-mysql-summary

```
       InnoDB Queue | 0 queries inside InnoDB, 0 queries in queue
 Oldest Transaction | 0 Seconds
   History List Len | 6
         Read Views | 1
   Undo Log Entries | 0 transactions, 0 total undo, 0 max undo
  Pending I/O Reads | 0 buf pool reads, 0 normal AIO, 0 ibuf AIO, 0 preads
 Pending I/O Writes | 0 buf pool (0 LRU, 0 flush list, 0 page); 0 AIO, 0 sync, 0
  log IO (0 log, 0 chkp); 0 pwrites
Pending I/O Flushes | 0 buf pool, 0 log
 Transaction States | 1xnot started
```

pt-mysql-summary

```
# MyISAM #####################################################
           Key Cache | 16.0k
            Pct Used | 20%
           Unflushed | 0%
# Security ###################################################
               Users | 4 users, 0 anon, 0 w/o pw, 0 old pw
       Old Passwords | OFF
# Binary Logging #############################################
             Binlogs | 0
          Zero-Sized | 0
          Total Size | 0.0k
       binlog_format | STATEMENT
    expire_logs_days | 7
         sync_binlog | 0
           server_id | 0
        binlog_do_db | 
    binlog_ignore_db | 
```

pt-mysql-summary

```
# Noteworthy Variables #######################################
Auto-Inc Incr/Offset | 1/1
default_storage_engine | 0
          flush_time | 0
        init_connect | 0
           init_file | 0
            sql_mode | 0
    join_buffer_size | 128k
    sort_buffer_size | 64k
    read_buffer_size | 256k
read_rnd_buffer_size | 256k
  bulk_insert_buffer | 0k
 max_heap_table_size | 16M
      tmp_table_size | 16M
  max_allowed_packet | 1M
        thread_stack | 256k
                 log | OFF
           log_error | /var/lib/mysql/huey.karwin.percona.com.err
        log_warnings | 1
    log_slow_queries | ON
log_queries_not_using_indexes | OFF
   log_slave_updates | OFF
```

pt-mysql-summary

```
# Configuration File #########################################
         Config File | Cannot autodetect, trying common locations
         Config File | /etc/my.cnf

[client]
port                                = 3306

[mysqld]
skip-name-resolve
skip-slave-start
expire-logs-days                    = 7
slow-query-log                      = 1
long-query-time                     = 20000
default-storage-engine              = innodb
innodb_buffer_pool_size             = 16M

. . .

# The End ####################################################
```

pt-stalk

http://www.percona.com/doc/percona-toolkit/pt-stalk.html

pt-stalk

- Wait for a problem to occur, then gather forensic data about MySQL and the system.
- Good when you want to diagnose a problem, but you don’t know when it happens.

What Data is Collected?

```
2012_11_30_17_49_13-df
2012_11_30_17_49_13-disk-space
2012_11_30_17_49_13-diskstats
2012_11_30_17_49_13-hostname
2012_11_30_17_49_13-innodbstatus1
2012_11_30_17_49_13-innodbstatus2
2012_11_30_17_49_13-interrupts
2012_11_30_17_49_13-lsof
2012_11_30_17_49_13-meminfo
2012_11_30_17_49_13-mutex-status1
2012_11_30_17_49_13-mutex-status2
2012_11_30_17_49_13-mysqladmin
2012_11_30_17_49_13-netstat
2012_11_30_17_49_13-netstat_s
2012_11_30_17_49_13-opentables1
2012_11_30_17_49_13-opentables2
2012_11_30_17_49_13-output
2012_11_30_17_49_13-pmap
2012_11_30_17_49_13-processlist
2012_11_30_17_49_13-procstat
2012_11_30_17_49_13-procvmstat
2012_11_30_17_49_13-ps
2012_11_30_17_49_13-slabinfo
2012_11_30_17_49_13-sysctl
2012_11_30_17_49_13-top
2012_11_30_17_49_13-trigger
2012_11_30_17_49_13-variables
2012_11_30_17_49_13-vmstat
2012_11_30_17_49_13-vmstat-overall
```

Ad Hoc Usage

```
$ pt-stalk --no-stalk

2012_11_30_17_49_13 Starting /usr/bin/pt-stalk --function=status --
   variable=Threads_running --threshold=25 --match= --cycles=0 --interval=0
   --iterations=1 --run-time=30 --sleep=0 --dest=/var/lib/pt-stalk --prefix=
   --notify-by-email= --log=/var/log/pt-stalk.log --pid=/var/run/pt-
   stalk.pid --plugin=

2012_11_30_17_49_13 Not stalking; collect triggered immediately
2012_11_30_17_49_13 Collect triggered
2012_11_30_17_49_13 Collector PID 2865
2012_11_30_17_49_13 Waiting up to 90 seconds for collectors to finish...
2012_11_30_17_50_43 Killing collector 2865
2012_11_30_17_50_43 Exiting because no more iterations
2012_11_30_17_50_43 /usr/bin/pt-stalk exit status 0
```

Background Usage

- Run pt-stalk run as a daemon and let it wait.

```
$ pt-stalk --daemonize ...event options...
$ tail -f /var/log/pt-stalk.log
2012_11_30_20_37_30 Check results: Threads_running=1, matched=no, cycles_true=0
2012_11_30_20_37_31 Check results: Threads_running=1, matched=no, cycles_true=0
2012_11_30_20_37_32 Check results: Threads_running=1, matched=no, cycles_true=0
2012_11_30_20_37_33 Check results: Threads_running=1, matched=no, cycles_true=0
. . .
```

Function, Variable & Threshold

- Function: the information source that pt-stalk polls for the trigger event.
  - status: SHOW GLOBAL STATUS
  - processlist: SHOW PROCESSLIST
  - *filename*: you can write a custom shell script

Function, Variable & Threshold

- Variable: what to watch in the information source.
  - status: watch the named status variable.
  - processlist: watch the named column.

Function, Variable & Threshold

- Threshold: if the watched value is greater than this threshold, a collect is triggered.
  - status: compare to the value of the given variable.
  - processlist: compare to the count of how many processes show the “match” value in the watched column.

Example: Watch Status

- Watch status and collect information when Threads_running is 25 or more.

```
$ pt-stalk --function status
    --variable Threads_running
    --threshold 25
```

Example: Watch Processlist

- Watch processlist and collect information when there are 10 or more processes with State=statistics

```
$ pt-stalk --function processlist
    --variable State
    --match statistics
    --threshold 10
```

Example: Watch Custom Script

- Watch processlist and collect information when there are 10 or more processes with State=statistics

```
$ cat > purge_not_working.sh
trg_plugin() {
  mysql $EXT_ARGV -E -e "SHOW ENGINE INNODB
  STATUS" | grep "^History list length" | awk
  '{print $4}'
}
$ pt-stalk --function purge_not_working.sh
    --threshold 200
```

pt-archiver

http://www.percona.com/doc/percona-toolkit/pt-archiver.html

pt-archiver

- Archive rows from a MySQL table into another table or a file.
- Works incrementally on chunks of rows.
- Deletes data from source safely.

Move Data

- Copy data from one MySQL instance to another, then delete from the source:

```
$ mysqldump -h huey -d imdb keyword
  | mysql -h dewey test

$ pt-archiver --progress 10000
  --source h=huey,D=imdb,t=keyword
  --dest h=dewey,D=test
  --where "1=1" --limit 1000 --commit-each

TIME                ELAPSED   COUNT

2012-12-03T01:37:36       0       0

2012-12-03T01:37:43       6   10000

. . .
```

Copy Data

- Copy data from one MySQL instance to another, but do not delete data from the source:

```
$ pt-archiver --progress 10000 --no-delete
  --source h=localhost,D=imdb,t=keyword
  --file '%t.csv'
  --where "1=1" --limit 1000 --commit-each
TIME                ELAPSED   COUNT
2012-12-03T01:28:27       0       0
2012-12-03T01:28:28       0   10000
2012-12-03T01:28:28       0   20000
. . .
```

Purge Data

- Delete orphan rows (slowly):

```
$ pt-archiver --progress 10000 --purge
  --source h=huey,D=imdb,t=person_info
  --where 'NOT EXISTS(SELECT * FROM name
  WHERE id=person_info.person_id)'
```

Limitations

- Destination table must exist.
- Archiving related data across tables is awkward.
    - You can use WHERE with subqueries, but not JOIN.

pt-query-digest

http://www.percona.com/doc/percona-toolkit/pt-query-digest.html

pt-query-digest

- Analyze query execution logs and generate a query report, filter, replay, or transform queries.
- If you learn only one tool in Percona Toolkit, make it this one!
- Capture all traffic in the slow query log... temporarily.

```
mysql> SET GLOBAL long_query_time=0;
. . . wait for traffic . . .
mysql> SET GLOBAL long_query_time=10;
```

Report Output (1)

```
$ pt-query-digest /var/lib/mysql/mysql-slow.log
```

```
# 20.3s user time, 160ms system time, 29.92M rss, 2.34G vsz
# Current date: Mon Aug 15 15:49:53 2011
# Hostname: huey.percona.com
# Files: shopsite-slow.log
# Overall: 88.68k total, 229 unique, 26.98 QPS, 245.51x concurrency ______
# Time range: 2011-08-15 16:00:43 to 16:55:30
# Attribute          total     min     max     avg     95%  stddev  median
# ============     ======= ======= ======= ======= ======= ======= =======
# Exec time        806989s      2s    160s      9s     30s     11s      3s
# Lock time             8s    21us     9ms    90us   159us    81us    76us
# Rows sent          2.35M       0 368.61k   27.84   49.17   1.95k    0.99
# Rows examine       2.74G       0 737.23k  32.44k 101.89k  41.45k  11.91k
# Query size        37.37M      42  16.77k  441.84  719.66  221.12  400.73
```

Report Output (2)

```
# Profile
# Rank Query ID           Response time     Calls R/Call   Apdx V/M   Item
# ==== ================== ================= ===== ======== ==== ===== ====
#    1 0x2C28E6666E1DB80F 521215.0518 64.6% 19450  26.7977 0.01  2.34 SELECT campaign_user
#    2 0xBAC856B3ED9D6303 145125.9331 18.0% 43282   3.3530 0.44  0.45 SELECT package_object
#    3 0x39997372657D28E2  16694.7209  2.1%  1705   9.7916 0.06  9.10 SELECT plug_form
#    4 0x3523ACB26E4C481A  14598.2371  1.8%  4740   3.0798 0.43  0.54 SELECT article_slideshow
#    5 0xA69DF0D16A7026B2  12565.8977  1.6%  1316   9.5486 0.03  1.48 SELECT campaign_user
#    6 0xB8356E351A6FFD21  12116.0409  1.5%   995  12.1769 0.02  1.58 SELECT category
#    7 0x8F72E45EC91BC0F9  11491.9428  1.4%  3436   3.3446 0.39  0.66 SELECT package_page
#    8 0x601559979824AADB   8302.8338  1.0%  2324   3.5726 0.41  0.44 SELECT template_item
#    9 0xCEB19656E4165CFD   5189.4078  0.6%   678   7.6540 0.05  1.31 SELECT article
#   10 0xCE5EE218C3751804   4890.2081  0.6%  1094   4.4700 0.27  1.18 SELECT article_resources
```

Report Output (3)

```
# Query 1: 6.77 QPS, 181.54x concurrency, ID 0x2C28E6666E1DB80F at byte 37195306
# This item is included in the report because it matches --limit.
# Scores: Apdex = 0.01 [1.0], V/M = 2.34
# Query_time sparkline: |      _^|
# Time range: 2011-08-15 16:00:43 to 16:48:34
# Attribute    pct   total     min     max     avg     95%  stddev  median
# ============ === ======= ======= ======= ======= ======= ======= =======
# Count         21   19450
# Exec time     64 521215s      2s     55s     27s     40s      8s     26s
# Lock time     29      2s    31us     9ms   120us   185us   112us   108us
# Rows sent      0  18.99k       1       1       1       1       0       1
# Rows examine  70   1.94G 102.83k 105.55k 104.70k 101.89k       0 101.89k
# Query size    19   7.22M     377     424  389.04  420.77   18.23  381.65
# String:
# Databases    shopsite
# Users        appuser
# Query_time distribution
#   1us
#  10us
# 100us
#   1ms
#  10ms
# 100ms
#    1s  #
#  10s+  ################################################################
```

Report Output (4)

```
# Tables
#    SHOW TABLE STATUS FROM `shopsite` LIKE 'campaign_user'\G
#    SHOW CREATE TABLE `shopsite`.`campaign_user`\G
# EXPLAIN /*!50100 PARTITIONS*/
            SELECT  COUNT(distinct email) AS count
            FROM    campaign_user
            WHERE   campaign_id     = '1'
                AND misc_2 IS NULL\G
```

Query Review

- Save each query type seen to a table.

```
$ pt-query-digest
  --review h=dewey,D=percona,t=query_review
  /var/lib/mysql/mysql-slow.log
```

- You can add notes to each query type.
- When you analyze next week's log, the report excludes previously reviewed queries.

Other Options

- Filtering queries.
- Grouping queries.
- Reading other sources of queries.
- Including query `EXPLAIN` reports.
- Recording query history for reviews, trending.
- Community tools for browsing and visualizing query review & query history data:
  - https://github.com/kormoc/Query-Digest-UI
  - https://github.com/box/Anemometer

Cautions

- Processing large logs can be resource-intensive.
  - Copy logs to another server to avoid overloading your production site.
- Reports contain real queries!
  - Can expose sensitive information.

pt-duplicate-key-checker

http://www.percona.com/doc/percona-toolkit/pt-duplicate-key-checker.html

pt-duplicate-key-checker

- Find duplicate indexes and foreign keys on MySQL tables.
  - MySQL permits you to create redundant keys.
  - Nearly every database has some.
  - Output is a series of `ALTER TABLE` statements ready to drop or reform duplicate indexes.

pt-duplicate-key-checker

```
$ pt-duplicate-key-checker
# tezt.media_pictures
#############################################################
# subject_node_id is a left-prefix of INDEX
# Key definitions:
# KEY `subject_node_id` (`subject_node_id`)
# KEY `INDEX` USING BTREE
(`subject_node_id`,`frame_id`,`file_id`,`source_id`),
# Column types:
#  `subject_node_id` int(11) unsigned default null
#  `frame_id` smallint(6) unsigned not null
#  `file_id` int(11) unsigned not null
#  `source_id` int(11) not null
# To remove this duplicate index, execute:
ALTER TABLE `tezt`.`media_pictures` DROP INDEX
`subject_node_id`;
```

pt-duplicate-key-checker

```
# Db.system_transaction
#############################################################
# FOREIGN KEY A (`table_id`) REFERENCES `Db`.`table` (`id`)
is a duplicate of FOREIGN KEY B (`table_id`) REFERENCES
`Db`.`table` (`id`)
# Key definitions:
# CONSTRAINT `A` FOREIGN KEY (`table_id`) REFERENCES
`table` (`id`)
# CONSTRAINT `B` FOREIGN KEY (`table_id`) REFERENCES
`table` (`id`)
# Column types:
#  `table_id` bigint(20) default null
# To remove this duplicate foreign key, execute:
ALTER TABLE `Db`.`system_transaction` DROP FOREIGN KEY `A`;
# MySQL uses the A index for this foreign key constraint
```

pt-table-checksum

http://www.percona.com/doc/percona-toolkit/pt-table-checksum.html

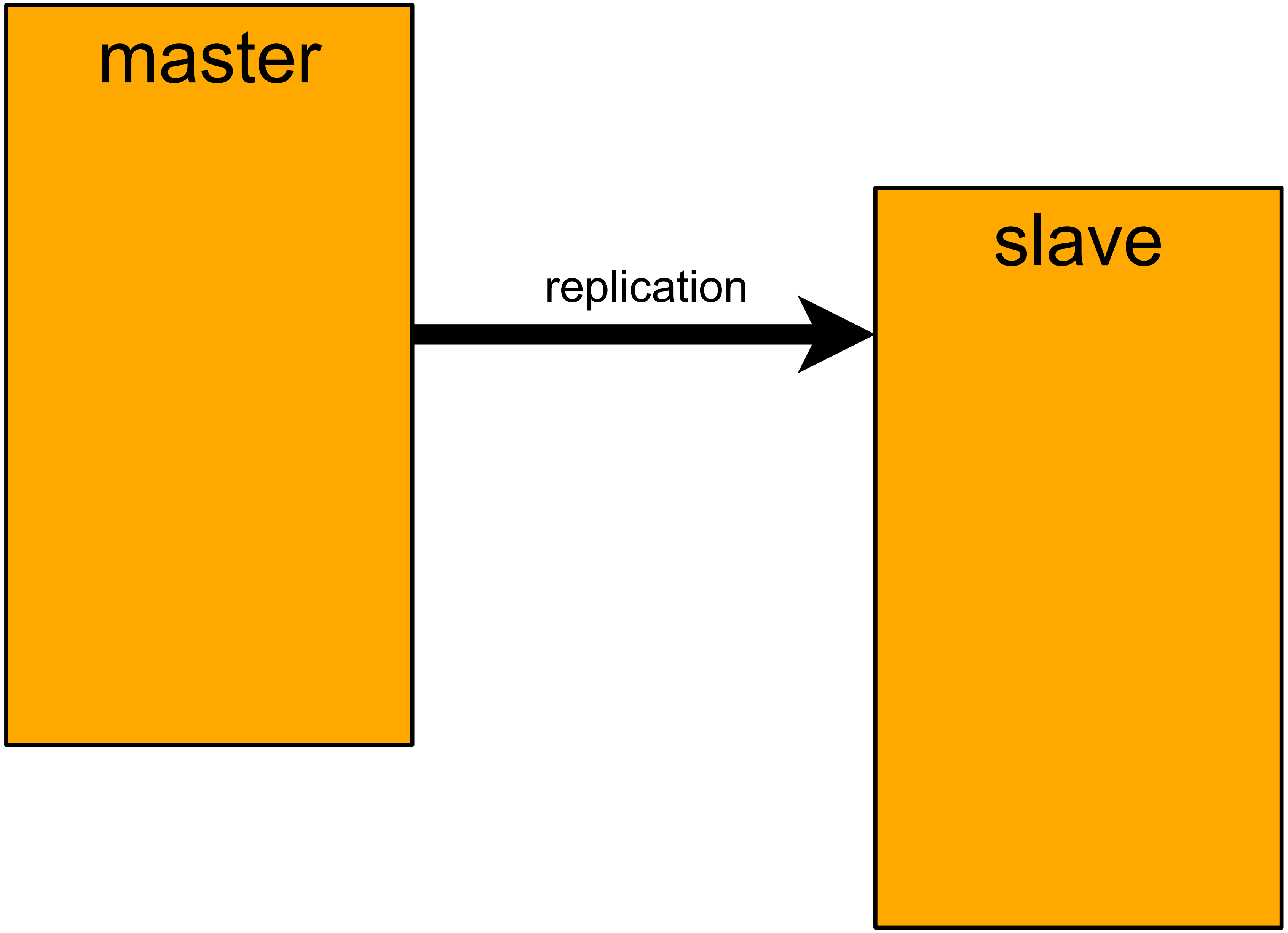
Data Drift

- MySQL slaves may not be perfect replicas.
  - Non-deterministic statements.
  - Out-of-band changes directly on the slave.
  - Slave may lag and fail to keep up.
  - No built-in checking.
  - Are you using a slave for backups or reporting?

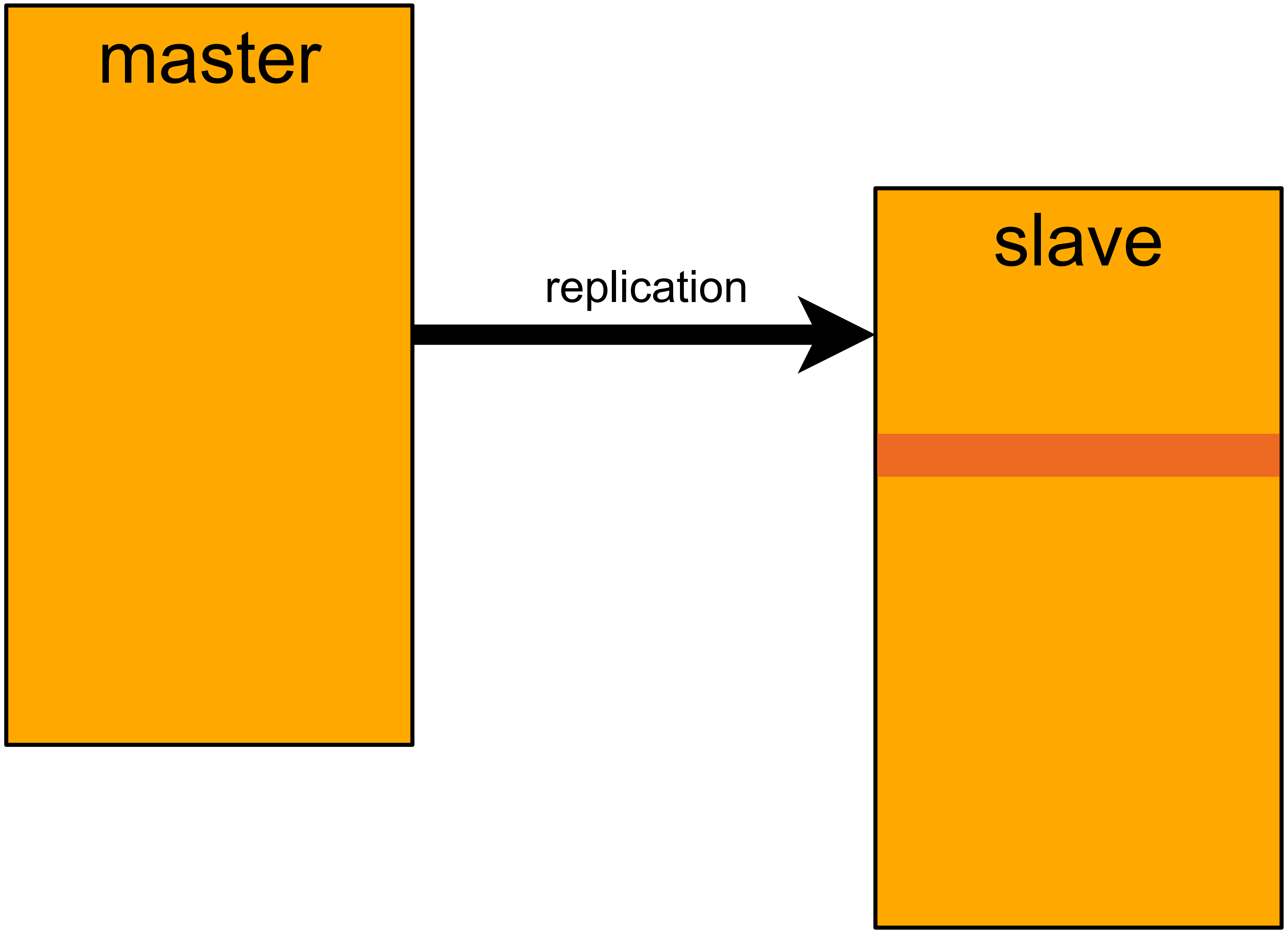
pt-table-checksum

- Perform an online replication consistency check, or checksum MySQL tables efficiently.
- This is the solution to *detect* data drift.
- Works by calculating checksums against “chunks” of rows.
- The calculation propagates to slaves.

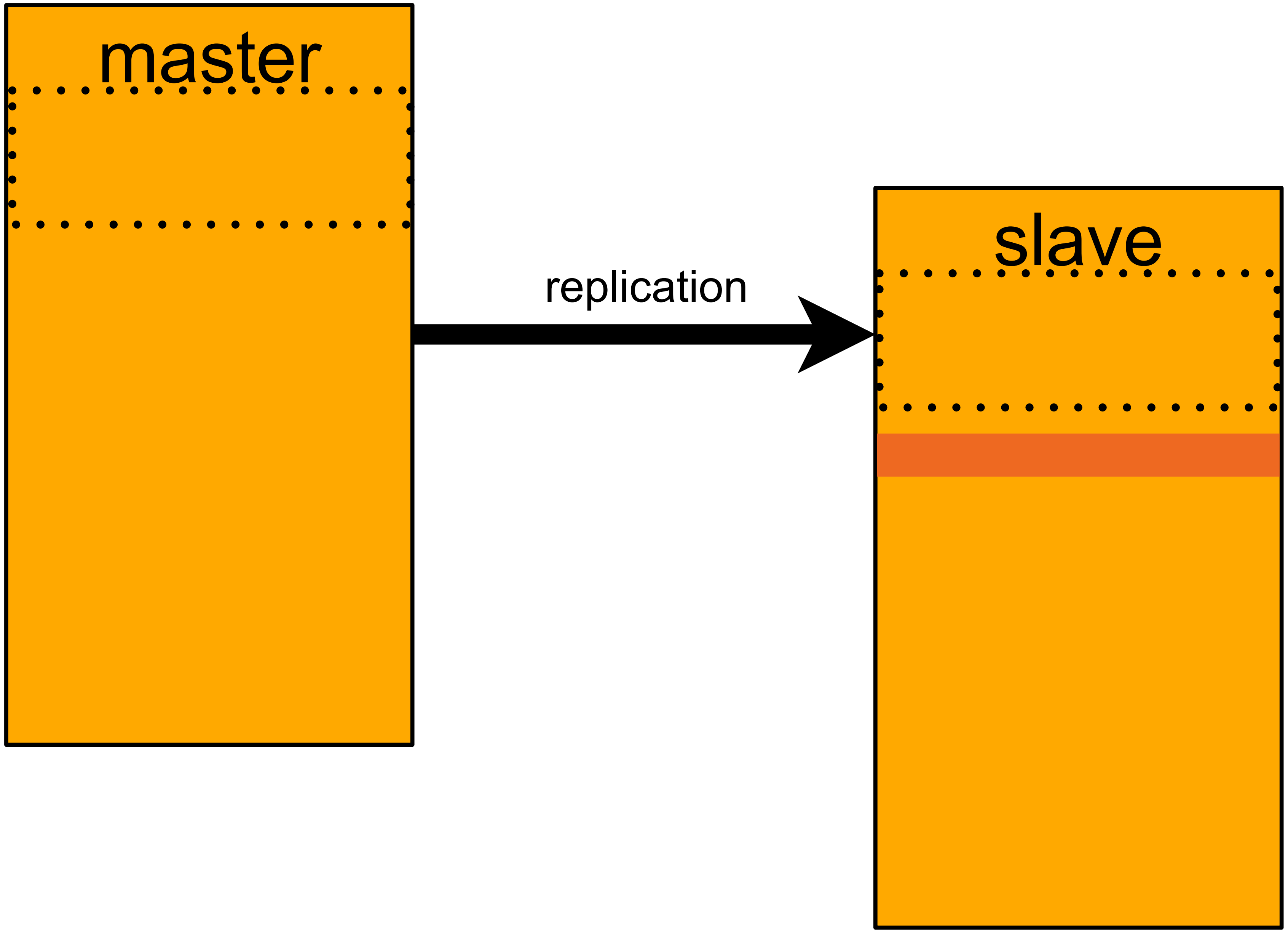
Visualize This



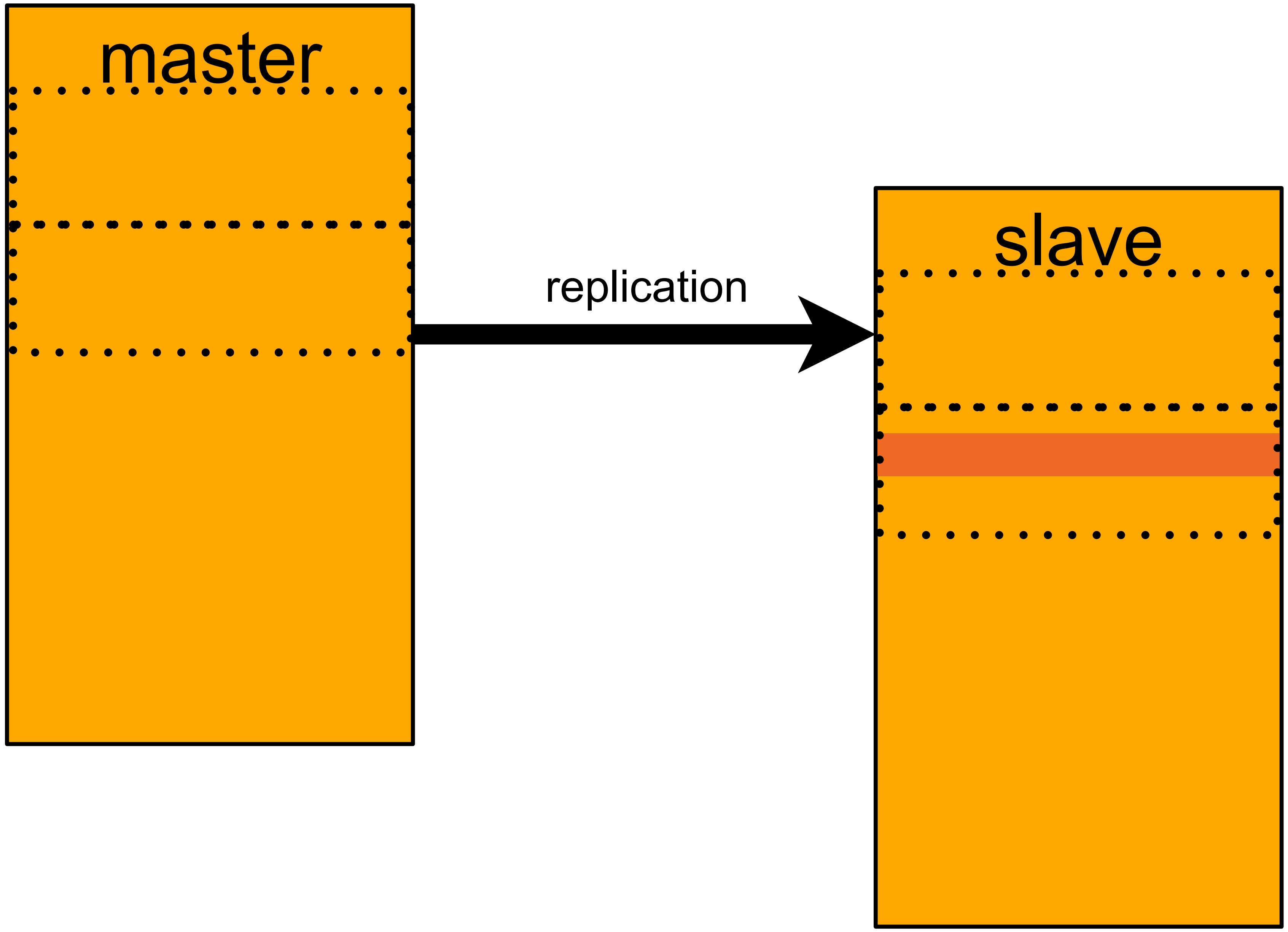
Visualize This



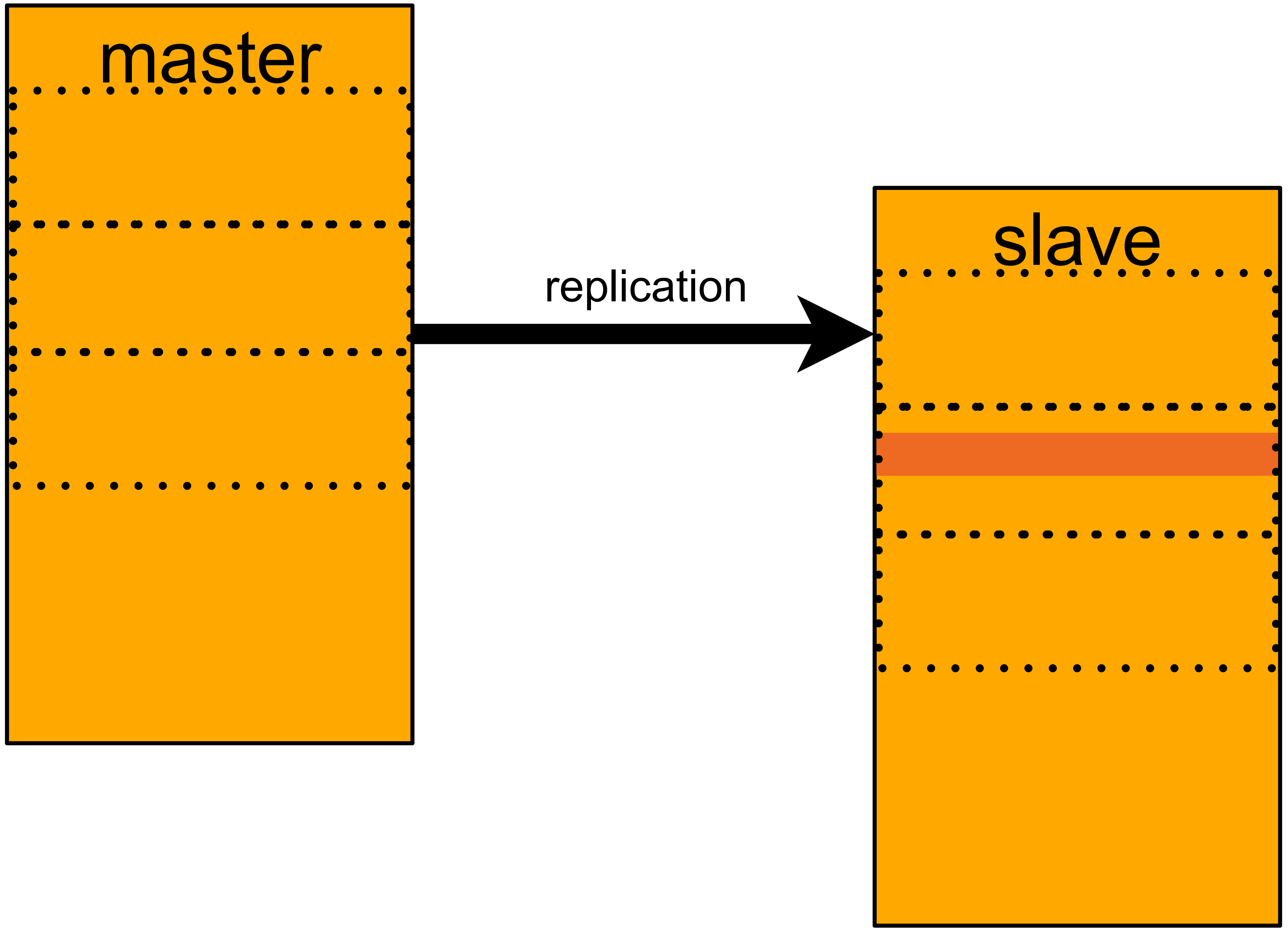
Visualize This



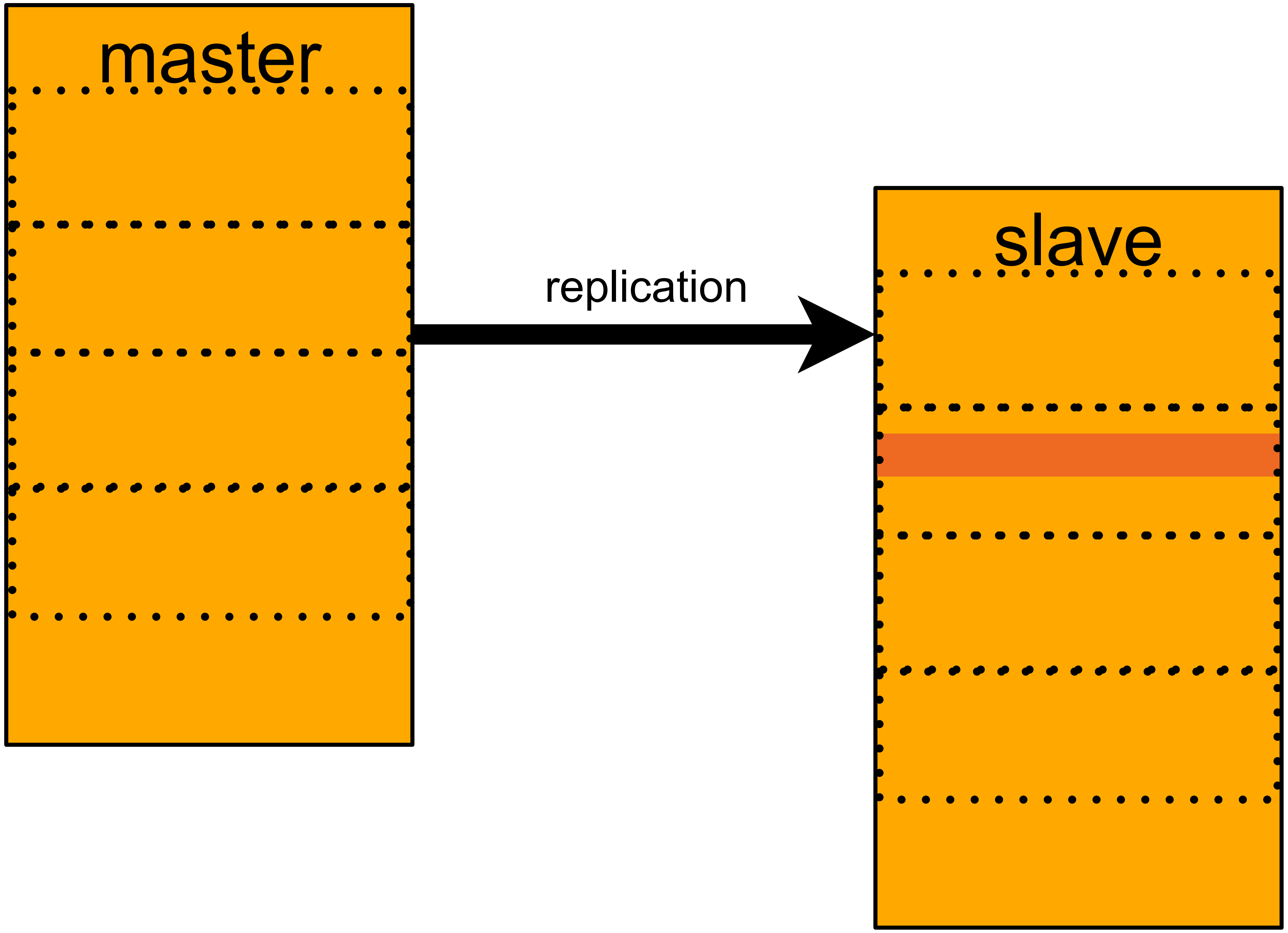
Visualize This



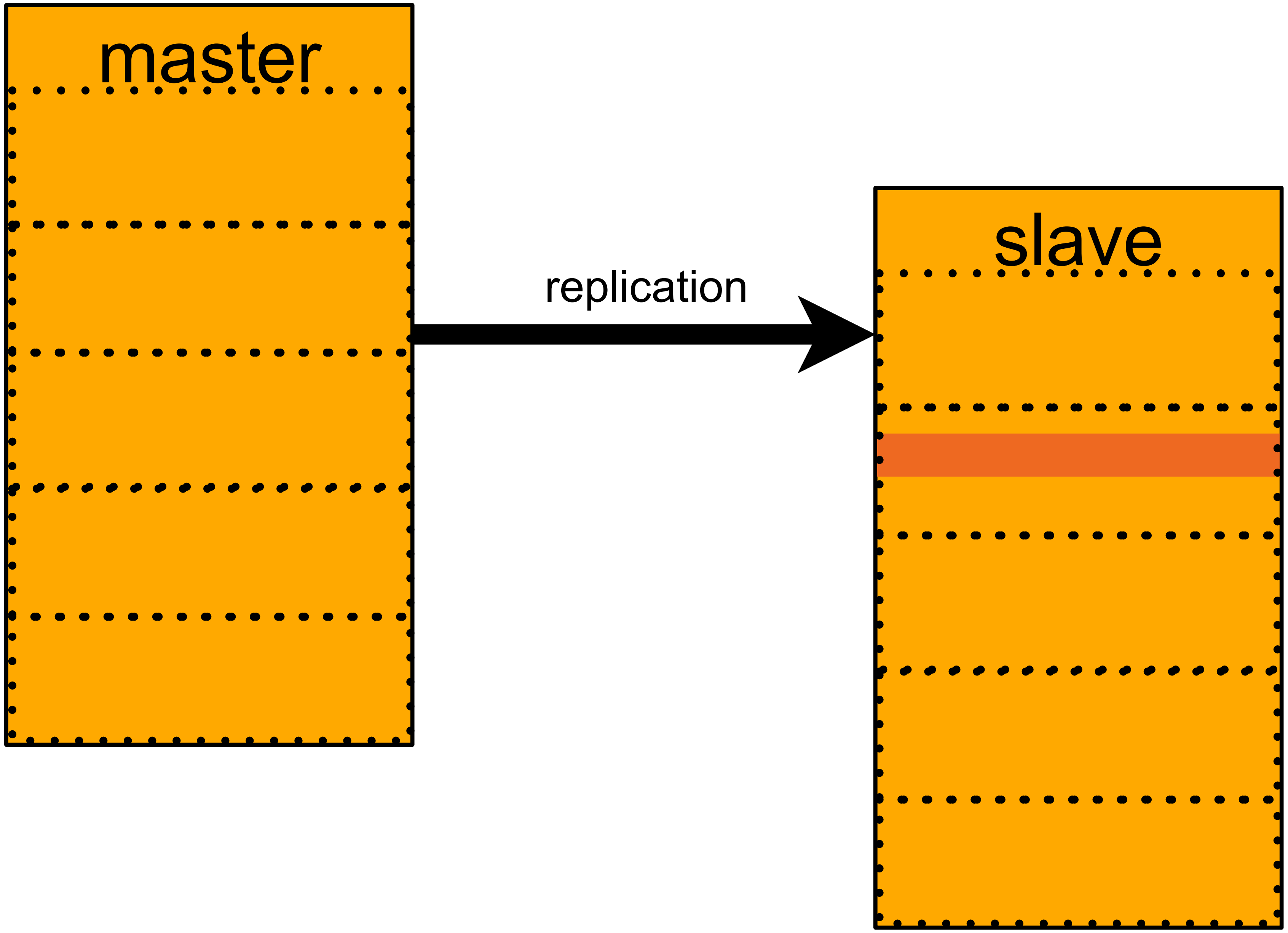
Visualize This



Visualize This



Visualize This



Example

```
$ pt-table-checksum
            TS ERRORS  DIFFS     ROWS  CHUNKS SKIPPED    TIME TABLE
12-01T11:00:13      0      0   633135       7       0   3.814 imdb.aka_name
12-01T11:00:15      0      0   290859       1       0   1.682 imdb.aka_title
Checksumming imdb.cast_info:  24% 01:34 remain
Checksumming imdb.cast_info:  48% 01:03 remain
Checksumming imdb.cast_info:  75% 00:28 remain
12-01T11:02:13      0      0 22187768     163       0 118.059 imdb.cast_info
12-01T11:02:25      0      0  2406561      20       0  12.292 imdb.char_name
12-01T11:02:25      0      0        4       1       0   0.123 imdb.comp_cast_type
12-01T11:02:27      0      0   241457       1       0   1.291 imdb.company_name
12-01T11:02:27      0      0        4       1       0   0.033 imdb.company_type
12-01T11:02:27      0      0    97304       1       0   0.492 imdb.complete_cast
12-01T11:02:27      0      0      113       1       0   0.079 imdb.info_type
12-01T11:02:28      0      0    87520       1       0   0.367 imdb.keyword
12-01T11:02:28      0      0        7       1       0   0.027 imdb.kind_type
12-01T11:02:28      0      0       18       1       0   0.030 imdb.link_type
12-01T11:02:37      0      0  1965016      15       0   9.142 imdb.movie_companies
Checksumming imdb.movie_info:  64% 00:16 remain
12-01T11:03:34      0      0  9748370      76       0  57.105 imdb.movie_info
12-01T11:03:38      0      0   934655       8       0   4.026 imdb.movie_info_idx
12-01T11:03:49      0      0  2776445      15       0  10.552 imdb.movie_keyword
12-01T11:03:52      0      0   922518       7       0   3.051 imdb.movie_link
12-01T11:04:07      0      0  2812743      25       0  15.817 imdb.name
12-01T11:04:29      0      0  2271731      22       0  21.495 imdb.person_info
12-01T11:04:29      0      0       12       1       0   0.015 imdb.role_type
12-01T11:04:39      0      0  1543719      17       0  10.189 imdb.title
```

Let’s Break It

- Delete 5% of data on the slave:

```
mysql> DELETE FROM title
  WHERE RAND()*100 < 5;

Query OK, 77712 rows affected (2.09 sec)
```

Re-check

```
$ pt-table-checksum --tables imdb.title

            TS ERRORS  DIFFS     ROWS  CHUNKS SKIPPED    TIME TABLE
12-03T05:04:26      0     14  1543719      16       0  10.512 imdb.title
```

Check the Slave(s)

```
mysql> SELECT db, tbl, SUM(this_cnt) AS total_rows, COUNT(*) AS chunks FROM
   percona.checksums WHERE (  master_cnt <> this_cnt  OR master_crc <>
   this_crc  OR ISNULL(master_crc) <> ISNULL(this_crc)) GROUP BY db, tbl;
+------+-------+------------+--------+
| db   | tbl   | total_rows | chunks |
+------+-------+------------+--------+
| imdb | title |    1466007 |     14 |
+------+-------+------------+--------+
```

pt-table-sync

http://www.percona.com/doc/percona-toolkit/pt-table-sync.html

pt-table-sync

- Synchronize MySQL table data efficiently.
- This is the solution to *correct* data drift.

Method 1: Sync Master to Slave(s)

```
$ pt-table-sync --verbose --execute --replicate percona.checksums huey

# Syncing via replication h=192.168.56.112
# DELETE REPLACE INSERT UPDATE ALGORITHM START    END      EXIT DATABASE.TABLE
#      0       47      0      0 Chunk     05:05:46 05:05:47 2    imdb.title
#      0      795      0      0 Chunk     05:05:47 05:05:49 2    imdb.title
#      0     5070      0      0 Chunk     05:05:49 05:06:01 2    imdb.title
#      0     6361      0      0 Chunk     05:06:01 05:06:16 2    imdb.title
#      0     6867      0      0 Chunk     05:06:16 05:06:36 2    imdb.title
#      0     7297      0      0 Chunk     05:06:36 05:06:55 2    imdb.title
#      0     7504      0      0 Chunk     05:06:55 05:07:13 2    imdb.title
#      0     7688      0      0 Chunk     05:07:13 05:07:34 2    imdb.title
#      0     7346      0      0 Chunk     05:07:34 05:07:52 2    imdb.title
#      0     7065      0      0 Chunk     05:07:52 05:08:10 2    imdb.title
#      0     6937      0      0 Chunk     05:08:10 05:08:27 2    imdb.title
#      0     6695      0      0 Chunk     05:08:27 05:08:43 2    imdb.title
#      0     6765      0      0 Chunk     05:08:43 05:09:00 2    imdb.title
#      0     1275      0      0 Chunk     05:09:00 05:09:04 2    imdb.title
```

Method 2: Sync Slave to Master

```
$ pt-table-sync --verbose --execute --sync-to-master h=dewey,D=imdb,t=title

# Syncing D=imdb,P=5528,h=127.0.0.1,p=...,t=title,u=root
# DELETE REPLACE INSERT UPDATE ALGORITHM START    END      EXIT DATABASE.TABLE
#      0   23097      0      0 Chunk     16:07:21 16:08:21 2    imdb.title
```

Method 3: Sync Two Hosts

- pt-table-sync won’t let you clobber a slave by syncing it to some host other than its master.

```
$ pt-table-sync --verbose --execute h=huey d=dewey --tables imdb.title

Can't make changes on h=dewey because it's a slave. See the documentation
   section 'REPLICATION SAFETY' for solutions to this problem. at /usr/bin/pt-
   table-sync line 10642.
```

Method 3: Sync Two Hosts

- Now let’s try again, after a RESET SLAVE.

```
$ pt-table-sync --verbose --execute h=huey h=dewey --tables imdb.title

# Syncing h=dewey
# DELETE REPLACE INSERT UPDATE ALGORITHM START    END      EXIT DATABASE.TABLE
#      0       0  30867      0 Chunk     13:33:27 13:35:28 2    imdb.title
```

pt-online-schema-change

http://www.percona.com/doc/percona-toolkit/pt-online-schema-change.html

pt-online-schema-change

- Perform online, non-blocking ALTER TABLE.
  - Captures concurrent updates to a table while restructuring.
  - Some risks and caveats exist; please read the manual and test carefully.

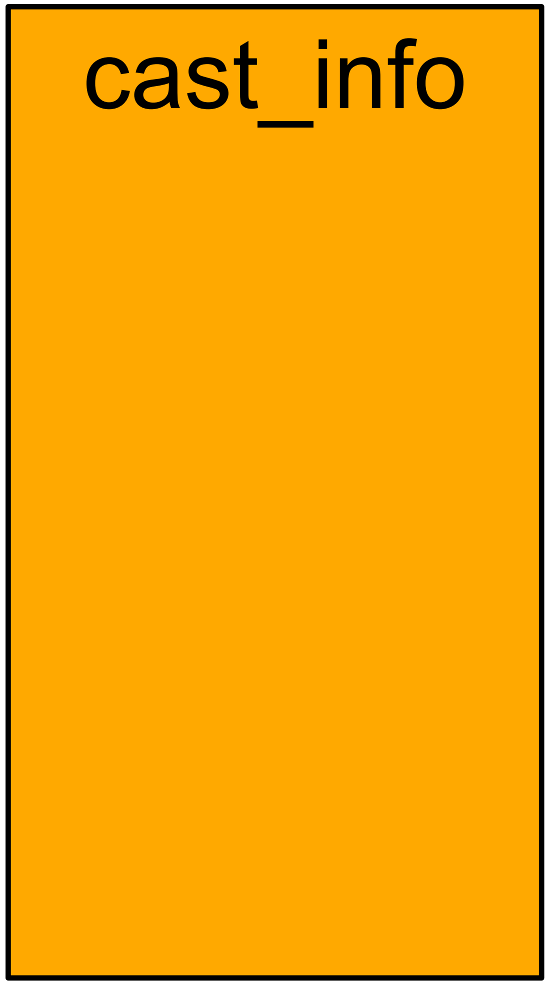
How MySQL Does ALTER TABLE

- Lock the table.
- Make a new, empty the table like the original.
- Modify the columns of the new empty table.
- Copy all rows of data from original to new table.
- Swap the old and new tables.
- Unlock the tables & drop the original.

How *pt-osc* Does ALTER TABLE

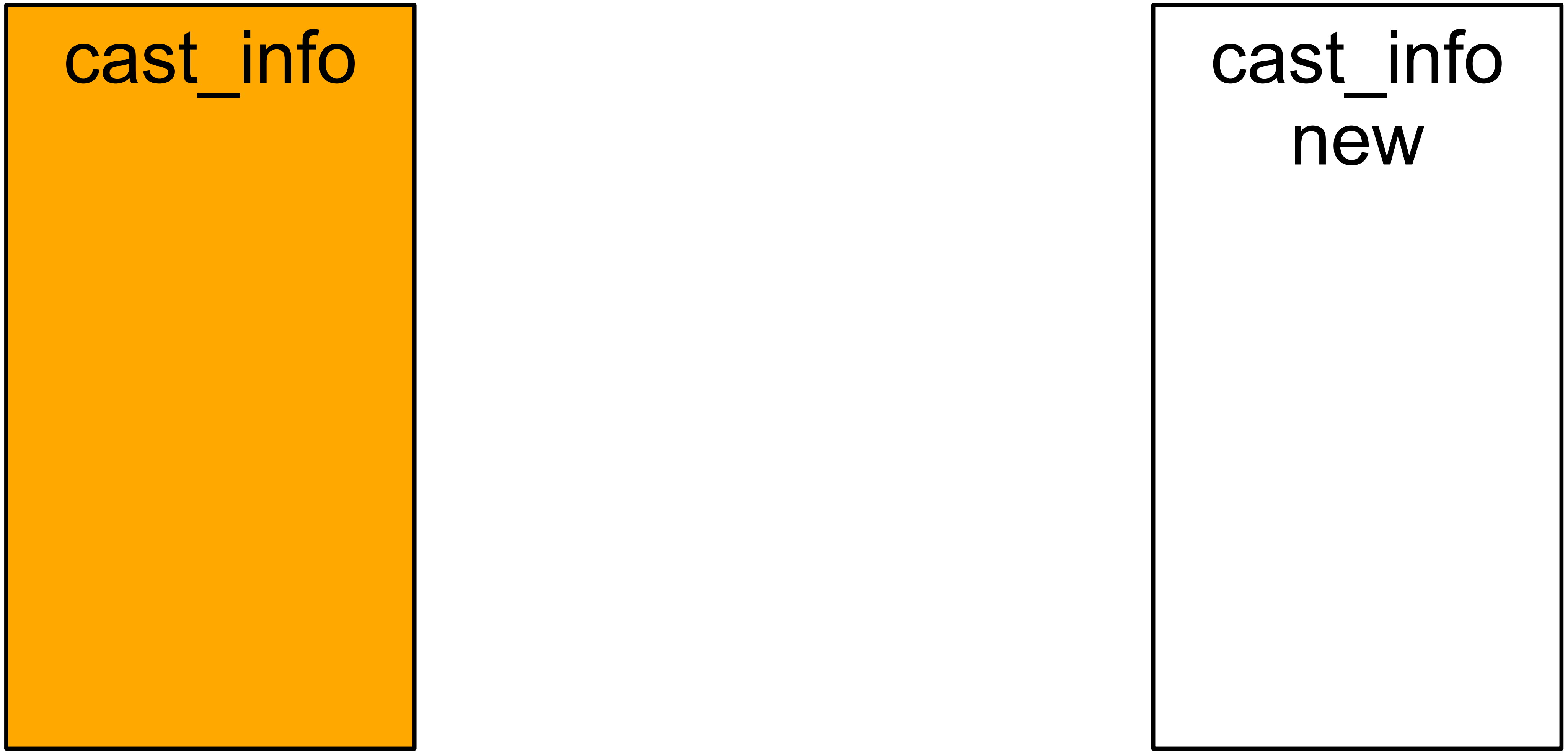
- ~~Lock the table.~~
- Make a new, empty the table like the original.
- Modify the columns of the new empty table.
- Copy all rows of data from original to new table.
  - Iterate over the table in chunks, in primary key order.
  - Use triggers to capture ongoing changes in the original, and apply them to the new table.
- Swap the tables, then drop the original.
- ~~Unlock the tables.~~

Visualize This



cast_info

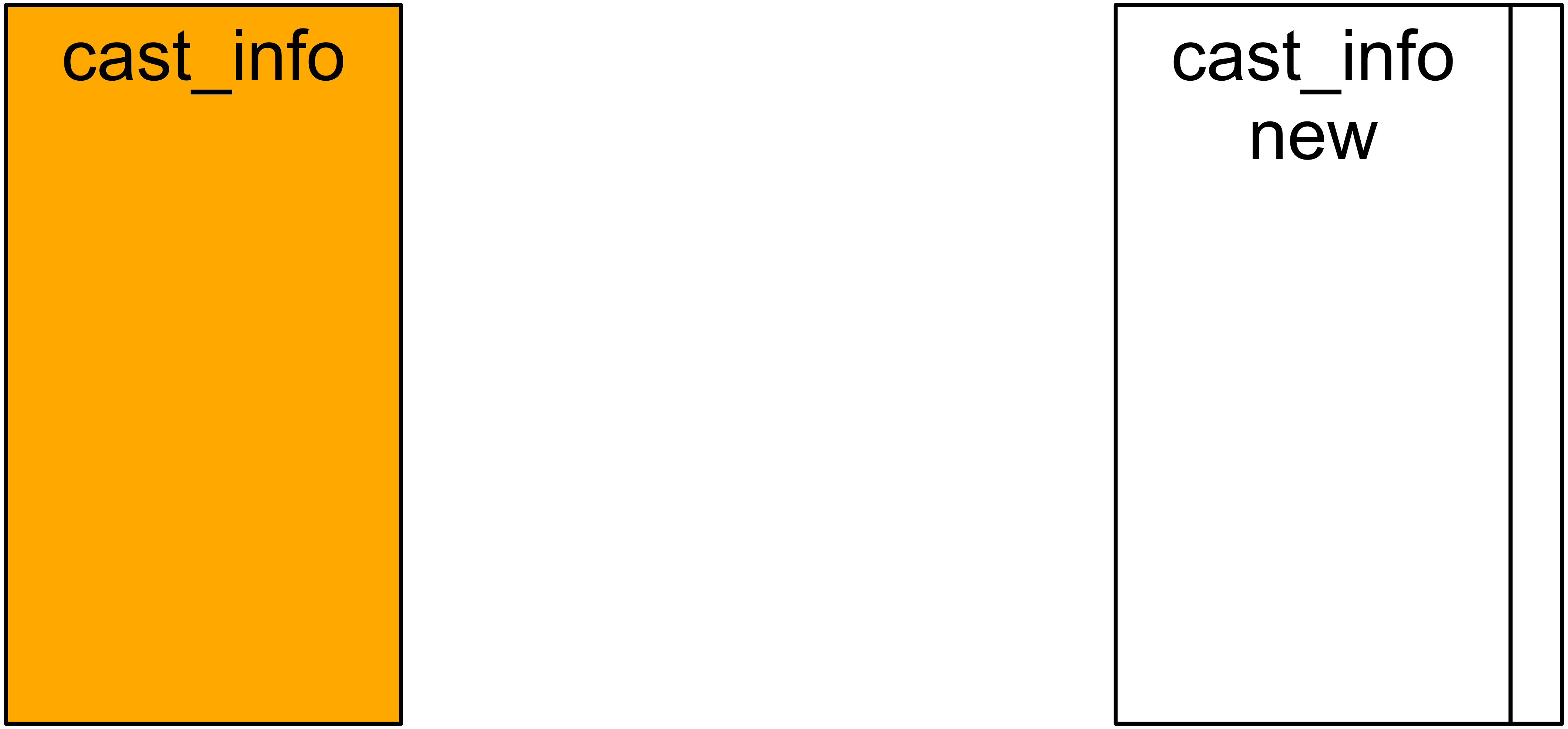
Visualize This



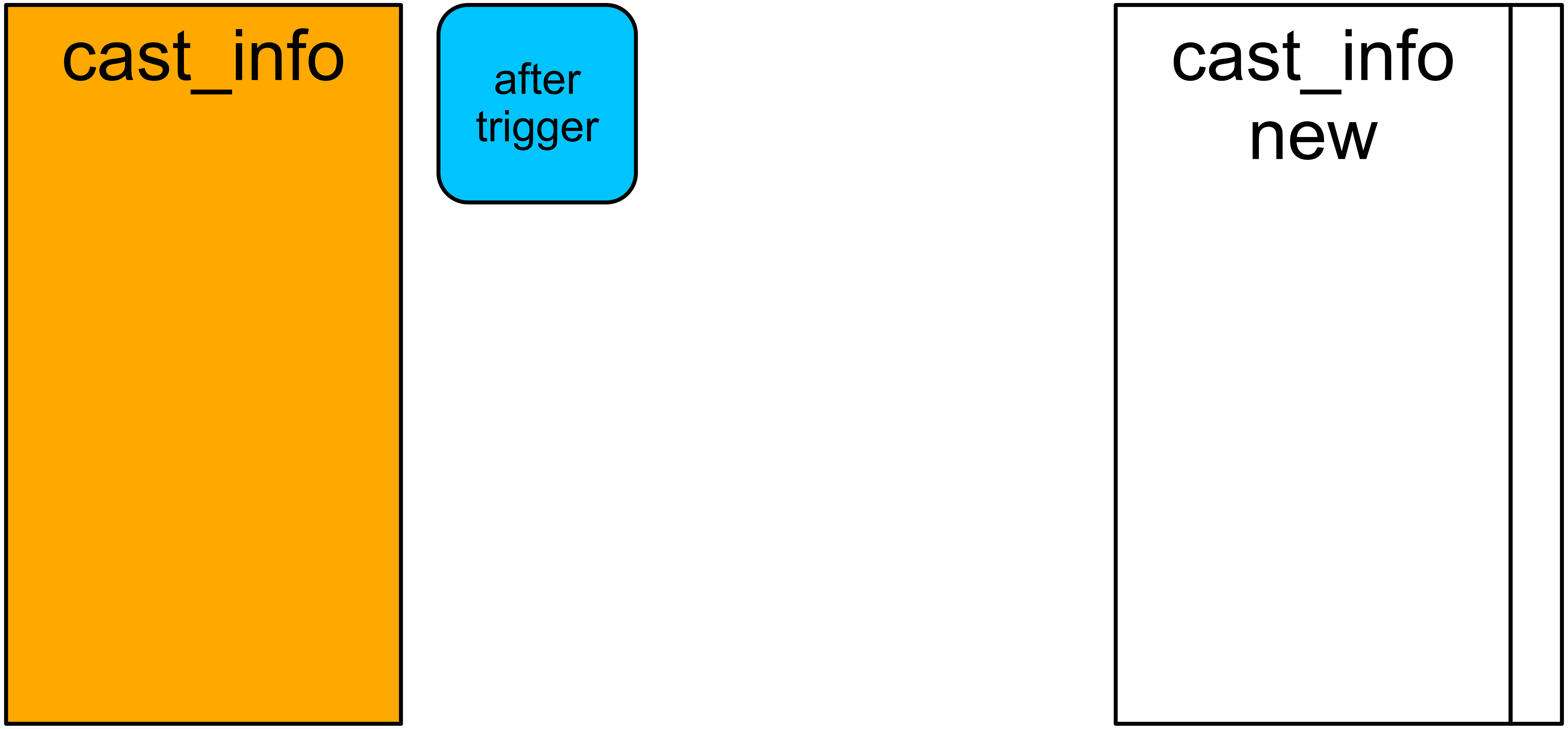
cast_info

cast_info new

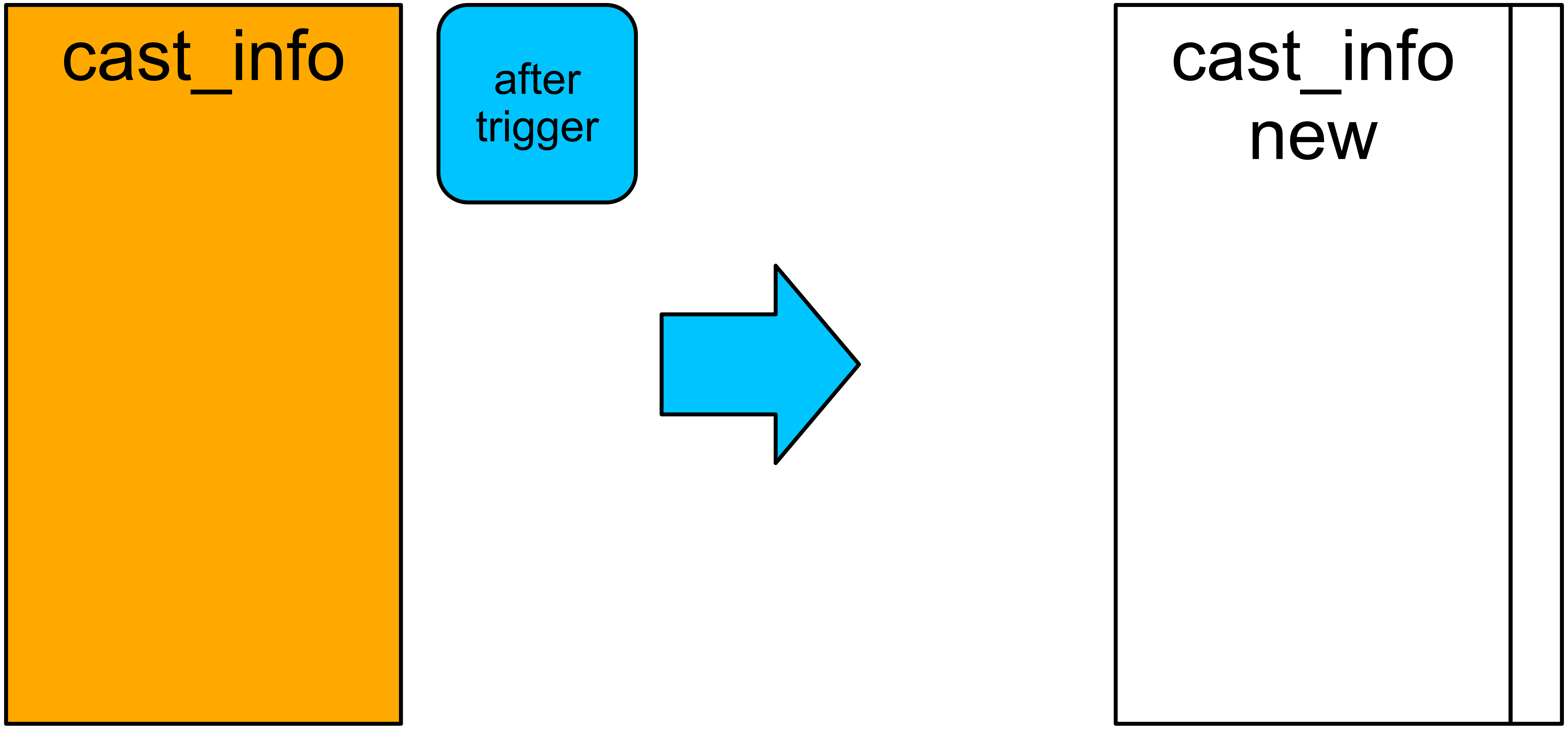
Visualize This



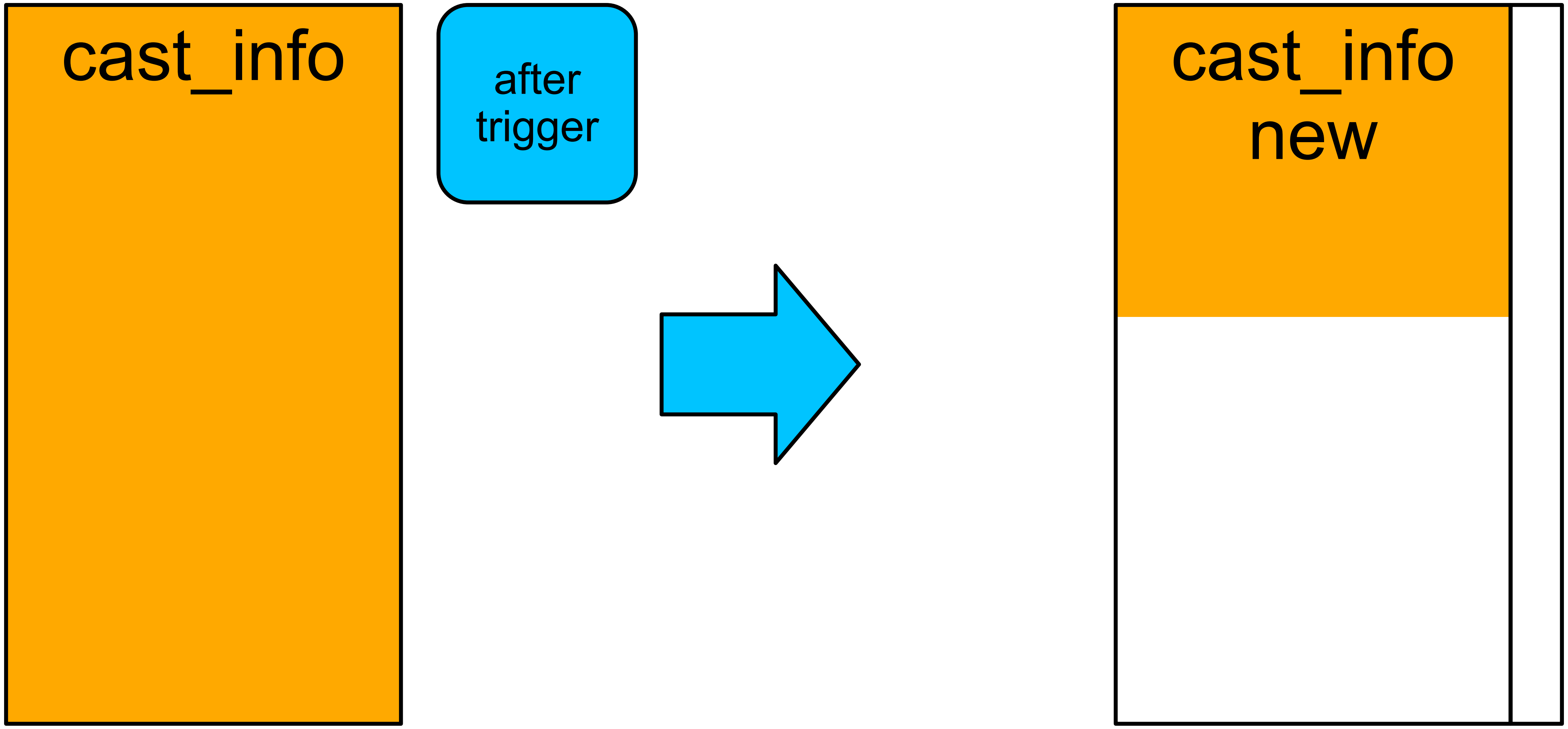
Visualize This



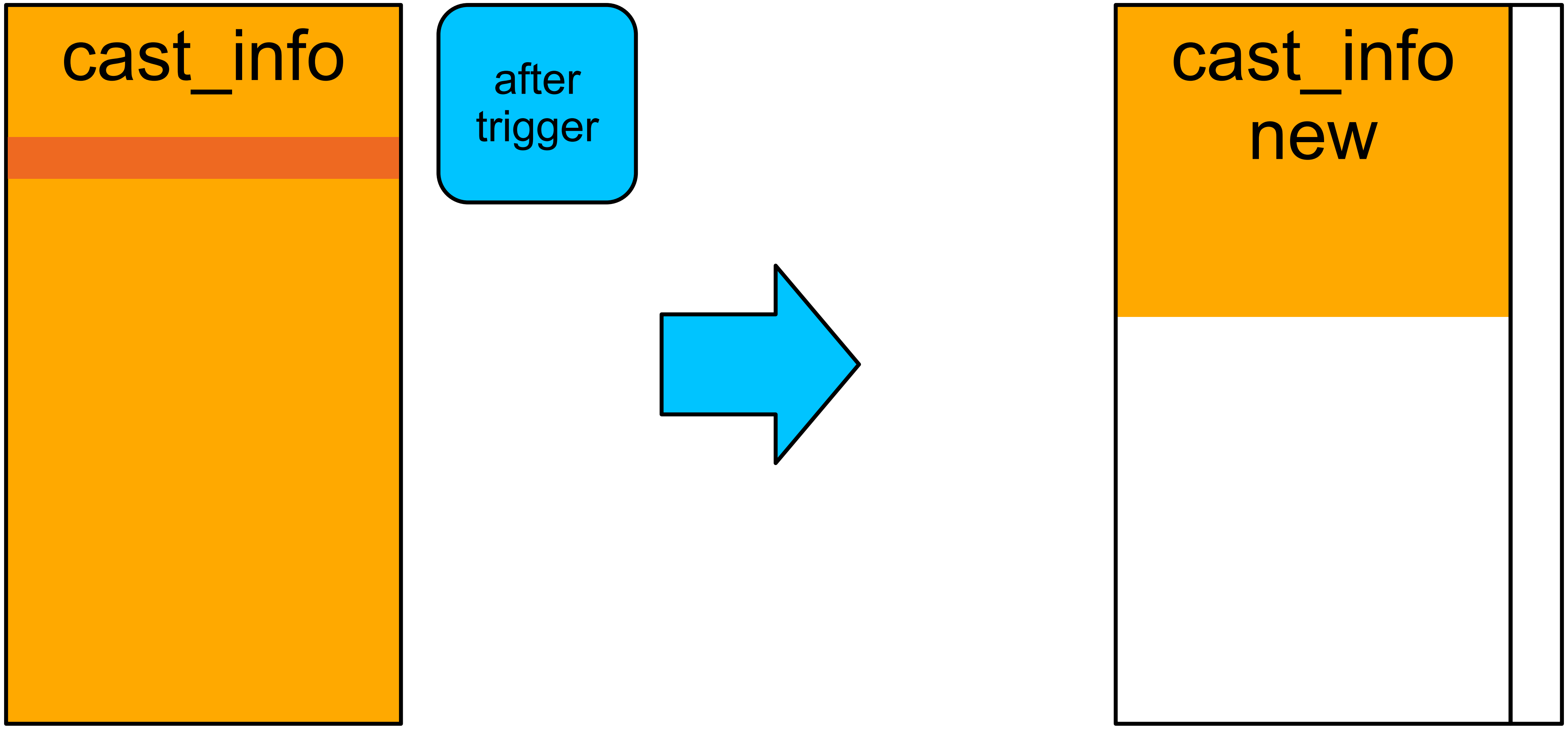
Visualize This



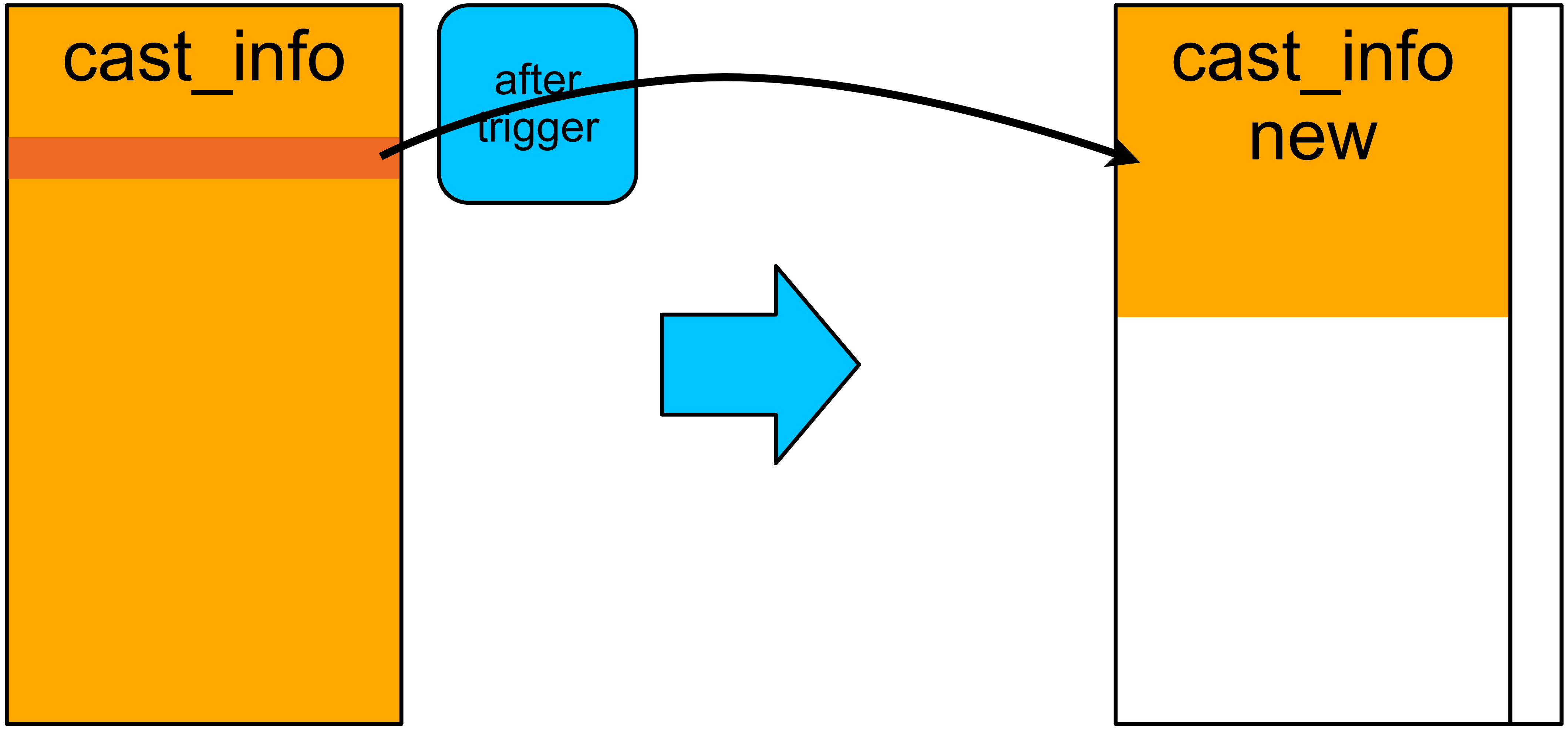
Visualize This



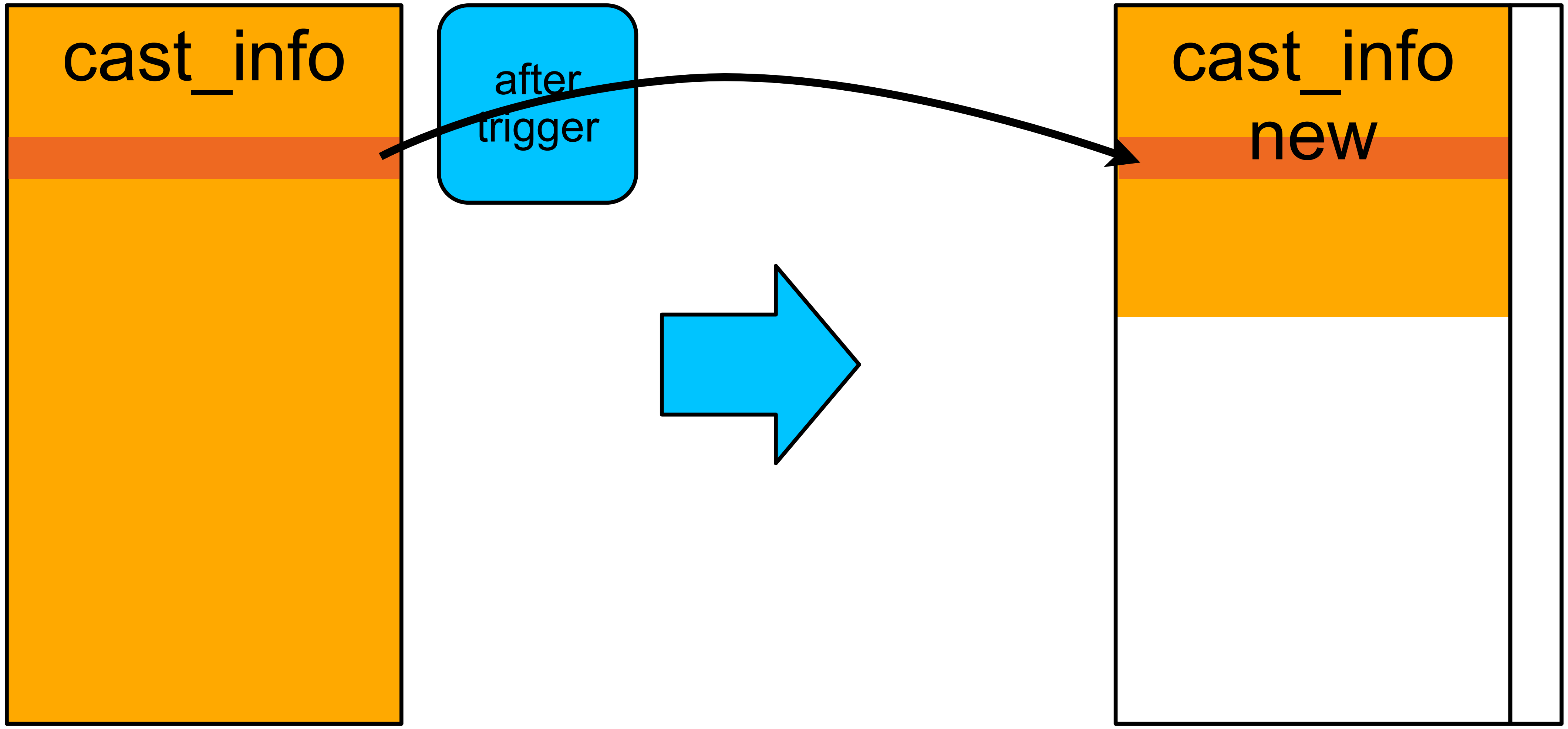
Visualize This



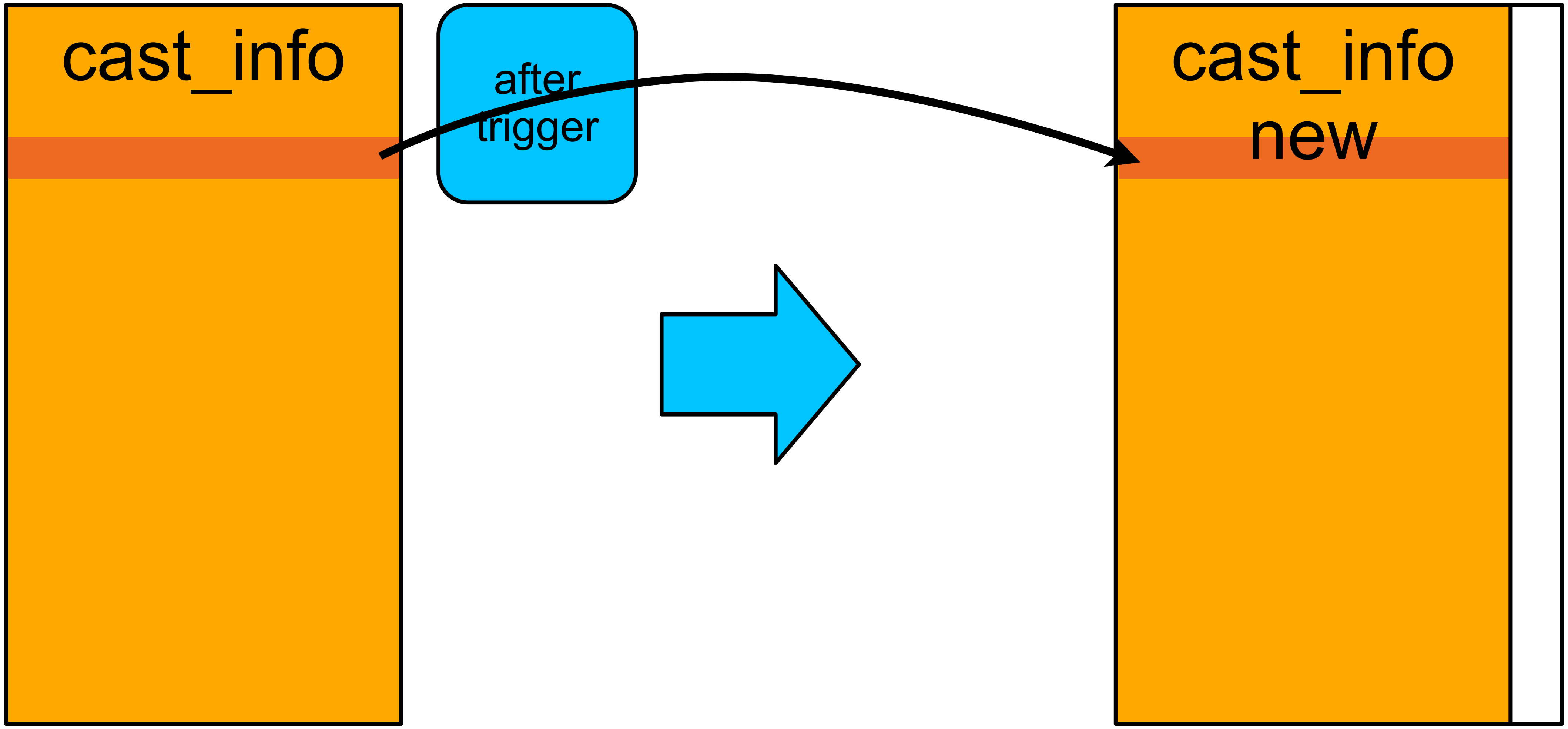
Visualize This



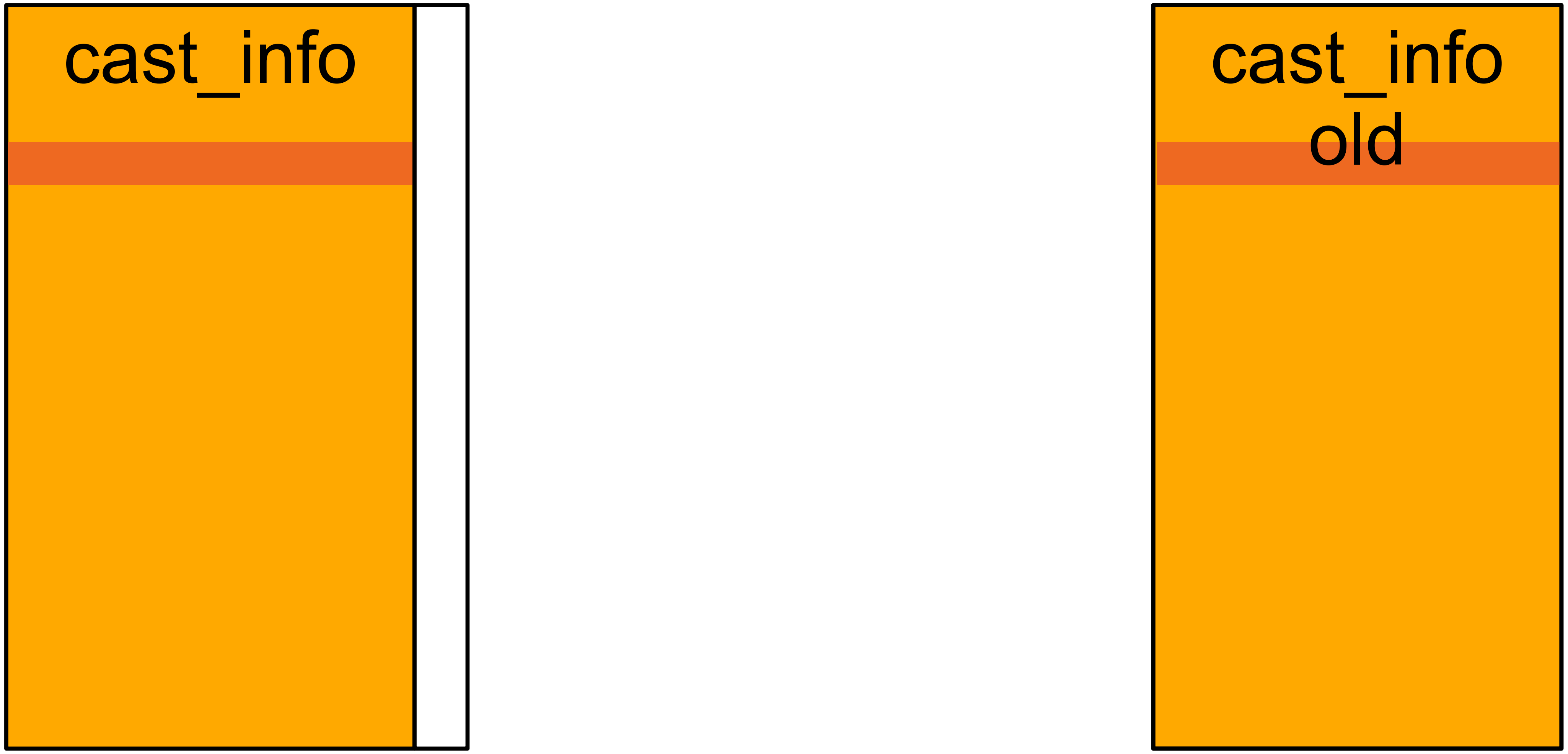
Visualize This



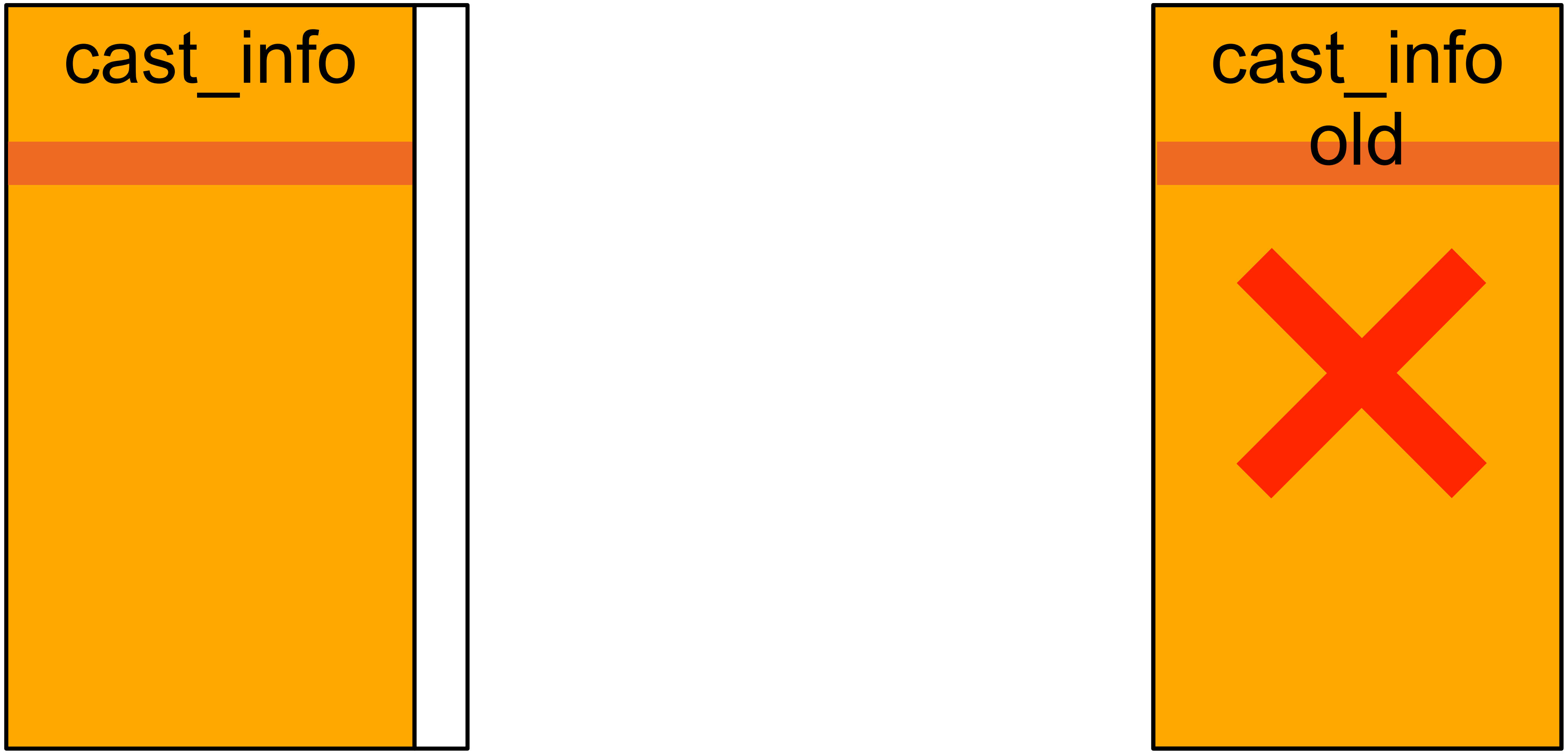
Visualize This



Visualize This



Visualize This



Example

```
$ pt-online-schema-change --alter "ADD COLUMN Dummy INT NOT NULL"
   h=localhost,D=imdb,t=cast_info

Exiting without altering `imdb`.`cast_info` because neither --dry-run nor --
   execute was specified.  Please read the tool's documentation carefully
   before using this tool.
```

Example

```
$ pt-online-schema-change --alter "ADD COLUMN Dummy INT NOT NULL"
   h=localhost,D=imdb,t=cast_info --execute
Altering `imdb`.`cast_info`...
Creating new table...
Created new table imdb._cast_info_new OK.
Altering new table...
Altered `imdb`.`_cast_info_new` OK.
Creating triggers...
Created triggers OK.
Copying approximately 22545051 rows...
Copying `imdb`.`cast_info`:  10% 04:05 remain
Copying `imdb`.`cast_info`:  19% 04:07 remain
Copying `imdb`.`cast_info`:  28% 03:44 remain
Copying `imdb`.`cast_info`:  37% 03:16 remain
Copying `imdb`.`cast_info`:  47% 02:47 remain
Copying `imdb`.`cast_info`:  56% 02:18 remain
Copying `imdb`.`cast_info`:  64% 01:53 remain
Copying `imdb`.`cast_info`:  73% 01:28 remain
Copying `imdb`.`cast_info`:  82% 00:55 remain
Copying `imdb`.`cast_info`:  91% 00:26 remain
Copied rows OK.
Swapping tables...
Swapped original and new tables OK.
Dropping old table...
Dropped old table `imdb`.`_cast_info_old` OK.
Dropping triggers...
Dropped triggers OK.
Successfully altered `imdb`.`cast_info`.
```

Self-Adjusting

- Copies rows in “chunks” which are sized dynamically by default.
- The tool throttles itself back if load increases too much or if any replication slaves are lagging.
- The tool tries to set its lock timeouts to let applications be more likely to succeed.

Limitations

- You can’t alter a table that already has triggers.
- You can’t add a column that is NOT NULL without also declaring a DEFAULT value.
- You can’t use replication filters.
- If the table is referenced by any foreign keys, you must choose a method to resolve them.

Why Shouldn’t I Use This?

- Is your table small enough that ALTER is already quick enough?
- Is your change already very quick, for example DROP KEY in InnoDB plugin?
- Will pt-online-schema-change cause the change to take too long or increase the load too much?
- Are you using MySQL 5.6, which supports online schema changes natively?

Cautions

- Execute pt-online-schema-change against a test instance first.
- Back up your data before you execute a change like this (and verify the backup is viable).

Top Nine Popular Tools

- pt-summary
- pt-mysql-summary
- pt-stalk
- pt-archiver
- pt-query-digest
- pt-duplicate-key-checker
- pt-table-checksum
- pt-table-sync
- pt-online-schema-change

Resources

- Product site: http://www.percona.com/software/percona-toolkit
- Percona support: http://www.percona.com/mysql-support
- Community support: https://groups.google.com/forum/?fromgroups#!forum/percona-discussion
- Bug tracker: https://bugs.launchpad.net/percona-toolkit

Webinars

- pt-stalk
  http://www.percona.com/webinars/2011-09-06-diagnosing-intermittent-mysql-problems
- pt-table-checksum
  http://www.percona.com/webinars/2012-01-18-verifying-replication-integrity-with-percona-toolkit
- pt-online-schema-change
  http://www.percona.com/webinars/2012-05-02-zero-downtime-schema-changes-in-mysql

There Are Many More Tools

pt-align

pt-config-diff

pt-deadlock-logger

pt-diskstats

pt-fifo-split

pt-find

pt-fingerprint

pt-fk-error-logger

pt-heartbeat

pt-index-usage

pt-ioprofile

pt-kill

pt-mext

pt-pmp

pt-query-advisor

pt-show-grants

pt-sift

pt-slave-delay

pt-slave-find

pt-slave-restart

pt-table-usage

pt-upgrade

pt-variable-advisor

pt-visual-explain

Percona Training for MySQL

PERCONA
TRAINING

Senior Industry Experts

In-Person and Online Classes

Custom Onsite Training

http://percona.com/training

Visit Our Sponsors

diamond

Clustrix

continuent

platinum

CouchBase

SkySQL

exhibitors

ospero
Global Hosting | Infrastructure | Platform Services

codership

Booking.com

Sphinx

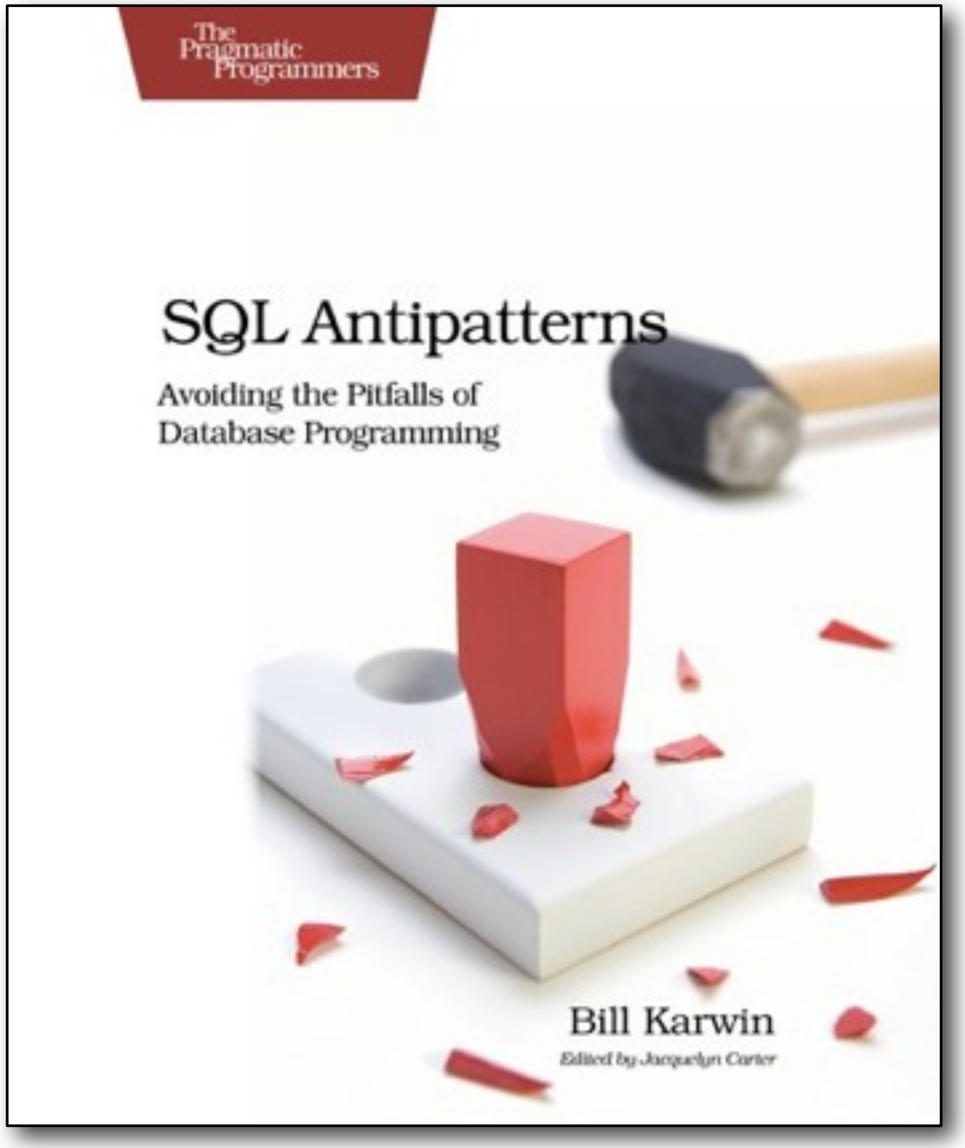
and the rest!

devart

MariaDB

New Relic

SQL Antipatterns


The Pragmatic Programmers
SQL Antipatterns
Avoiding the Pitfalls of Database Programming
Bill Karwin
Edited by Jacquelyn Carter

http://www.pragprog.com/titles/bksqla/

License and Copyright