



Inside the PrimeBase XT Storage Engine

MySQL Conference & Expo 2008

Paul McCullagh

PrimeBase Technologies GmbH

www.primebase.org





Contents

- Design & How it Works
- Applications of the Design: SSD
- Future of PBXT: HA Solutions



What is PrimeBase XT?

- A pluggable storage engine for MySQL 5.1+
- Transactional, ACID compliant (v1.0+)
- Open source (GPL), community project
- Designed and built specifically for MySQL
- Developed by PrimeBase Technologies:
<http://www.primebase.org>
- Hosted by Sourceforge.net:
<http://sourceforge.net/projects/pbxt>



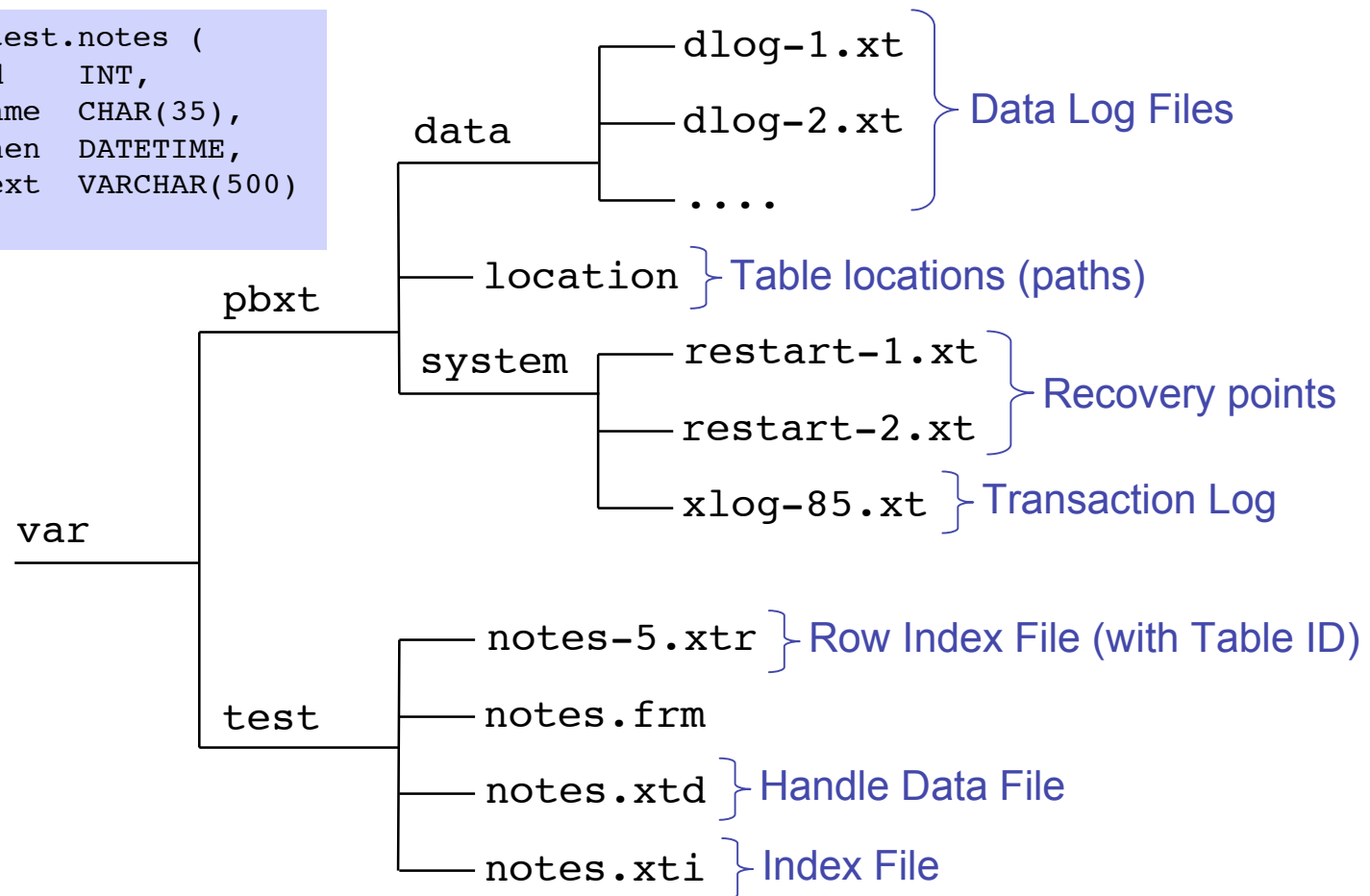
Design Principles

- MVCC, all versions are stored on disk.
- Writes sequentially/write once (log-based).
- Never updates in place.
- No undo, non-committed data is garbage (collected by background threads).
- File-per-table, no table spaces.



Disk Structure (v1.0.01)

```
CREATE TABLE test.notes (
    n_id      INT,
    n_name    CHAR(35),
    n_when    DATETIME,
    n_text    VARCHAR(500)
) ENGINE=PBXT;
```

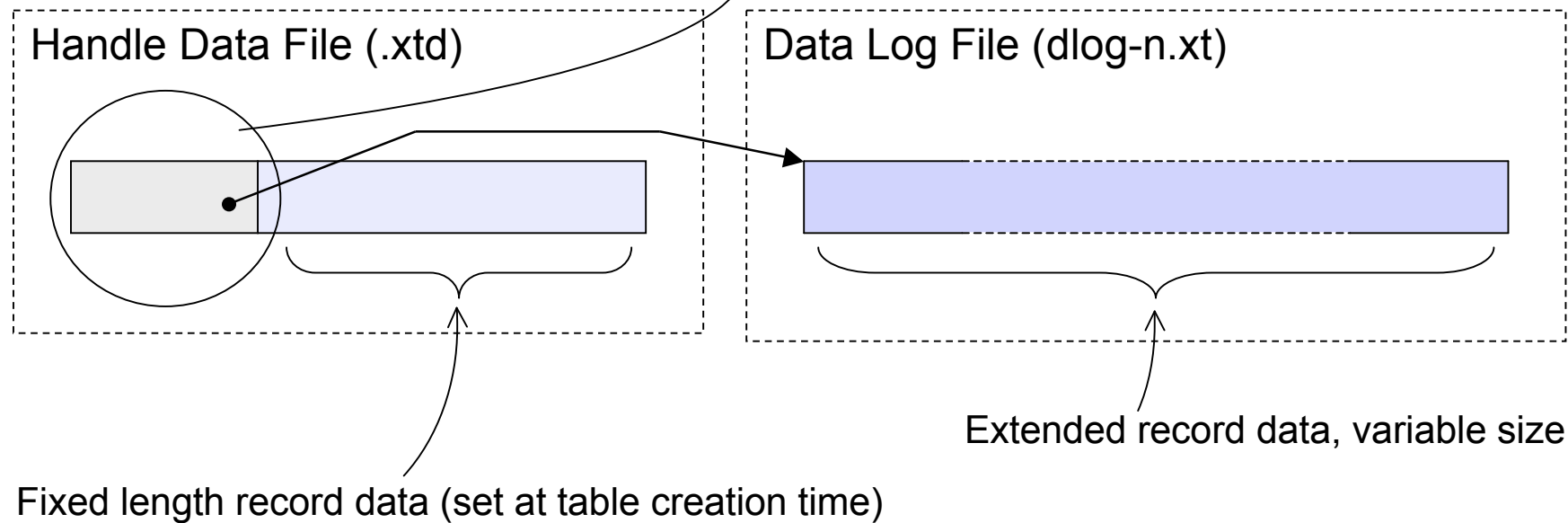




Record Structure

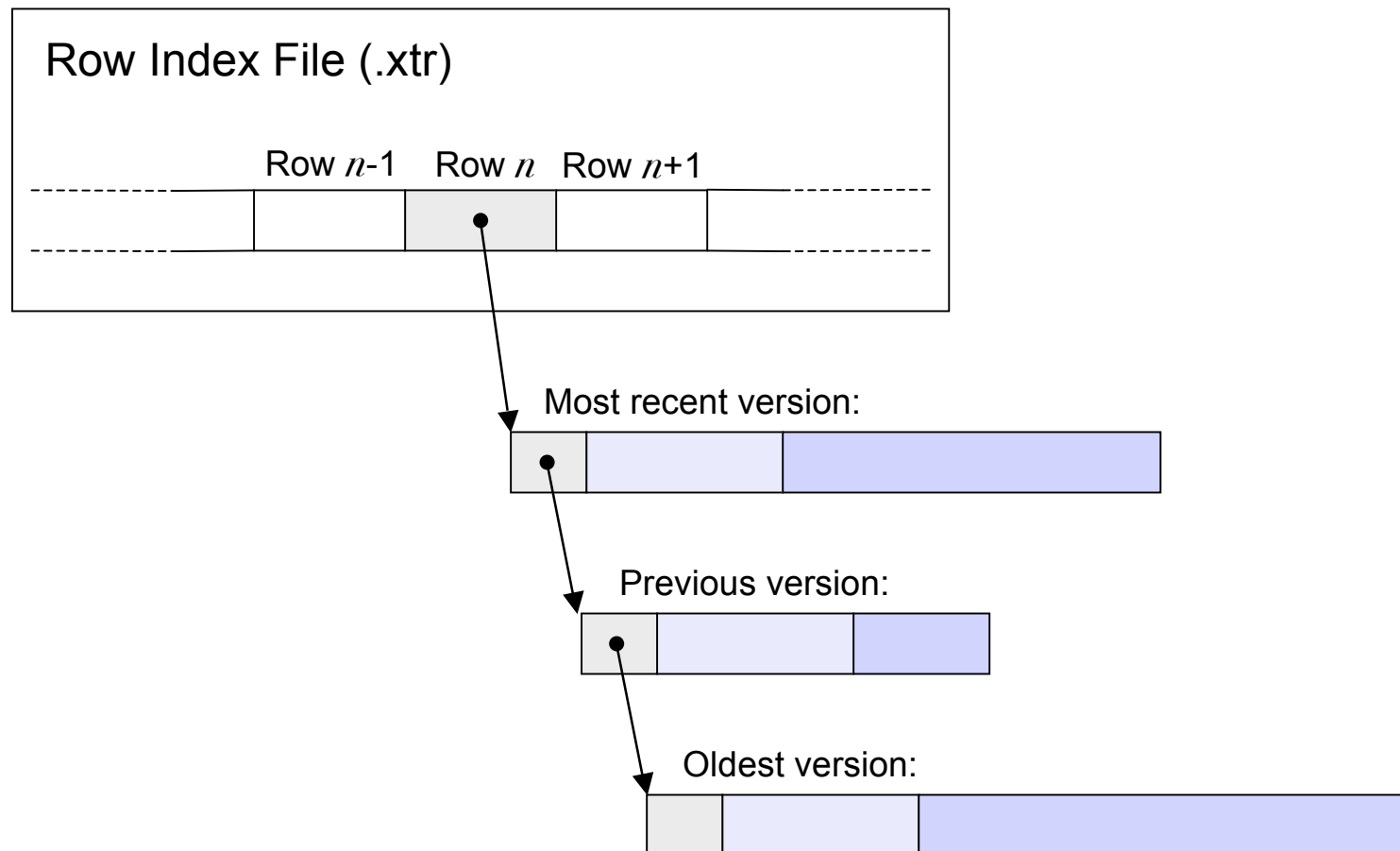
Control Data (14 - 26 bytes):

Status	Prev. version	Xaction ID	Row ID	Ext. Data Ref
--------	---------------	------------	--------	---------------



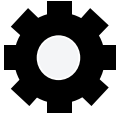
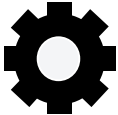


Row Structure



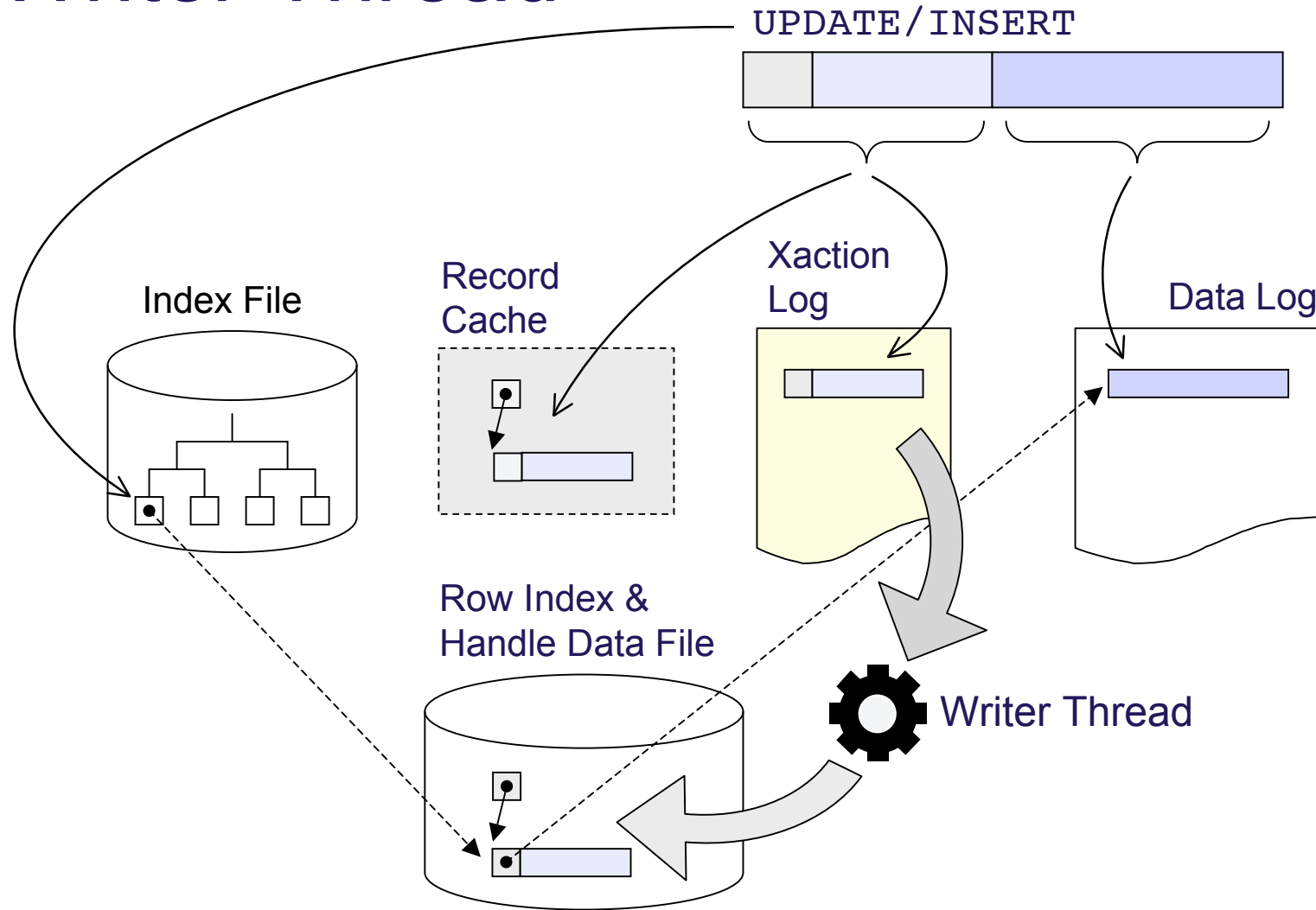


Background Threads

-  Writer
-  Sweeper
-  Compactor
-  Checkpointer

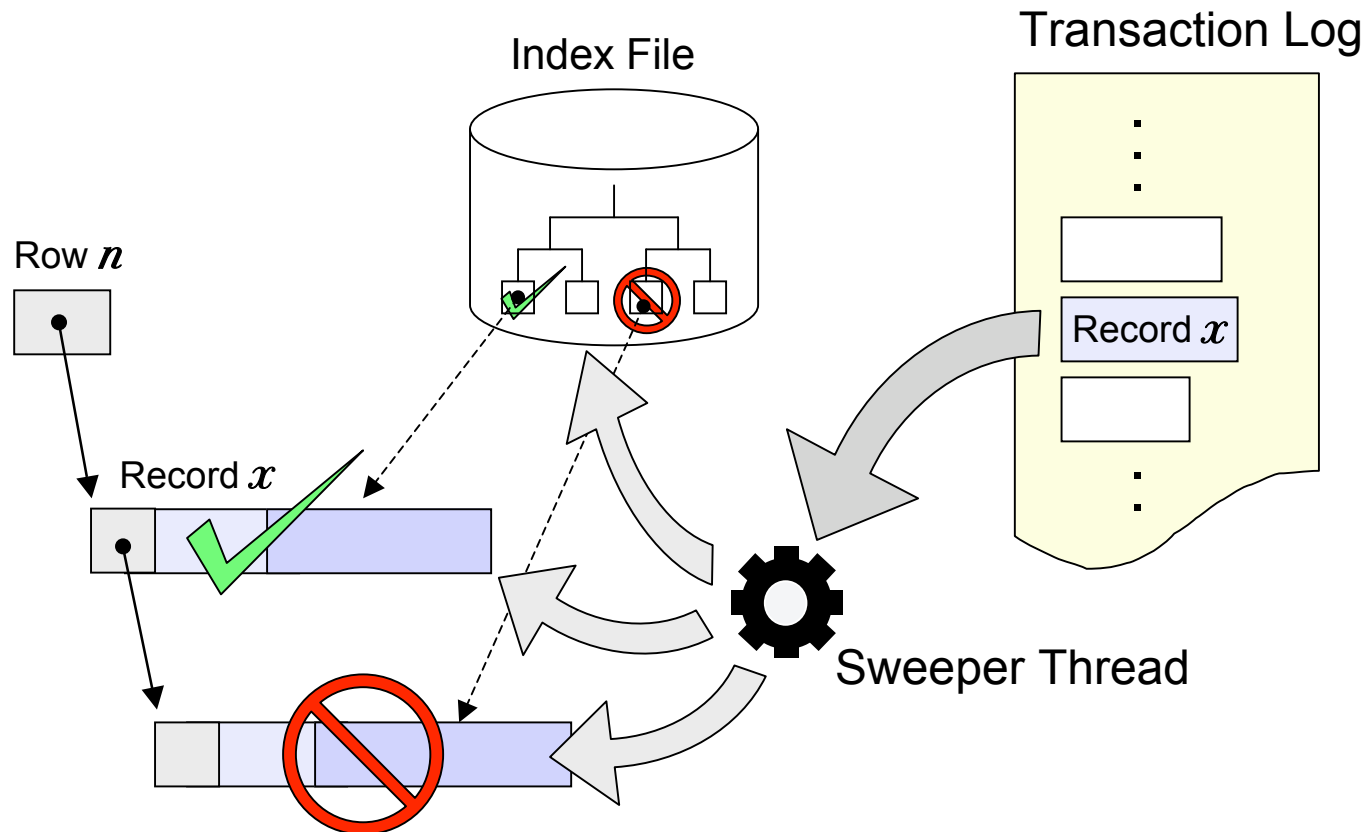


Writer Thread



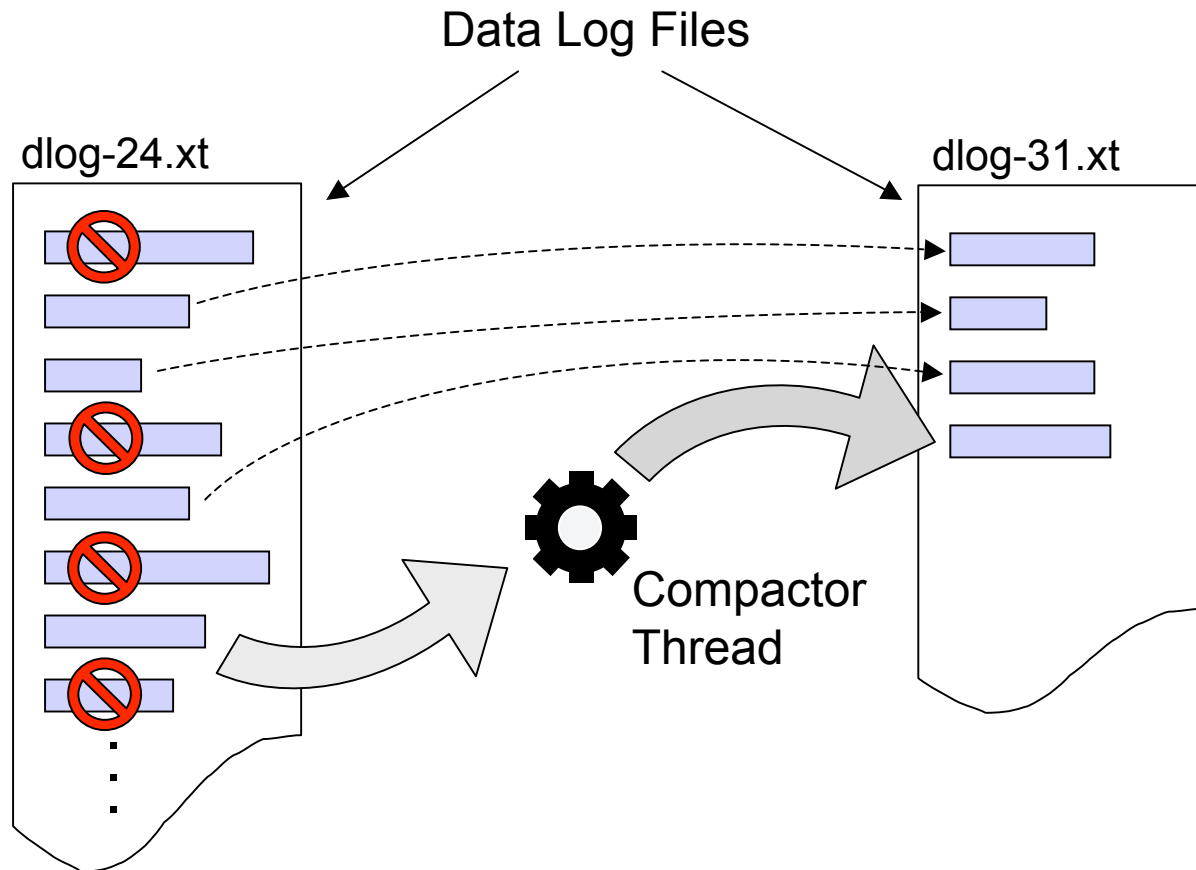


Sweeper Thread



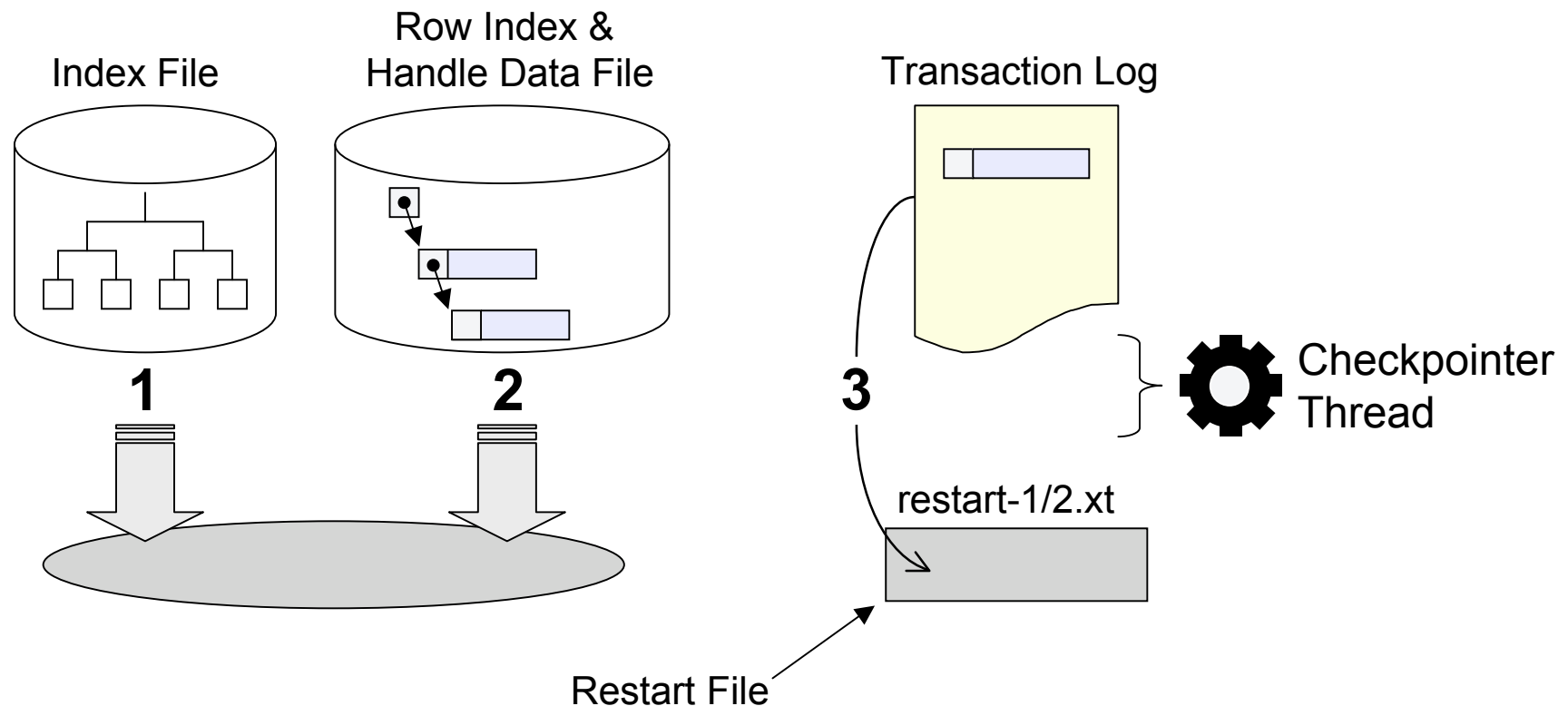


Compactor Thread



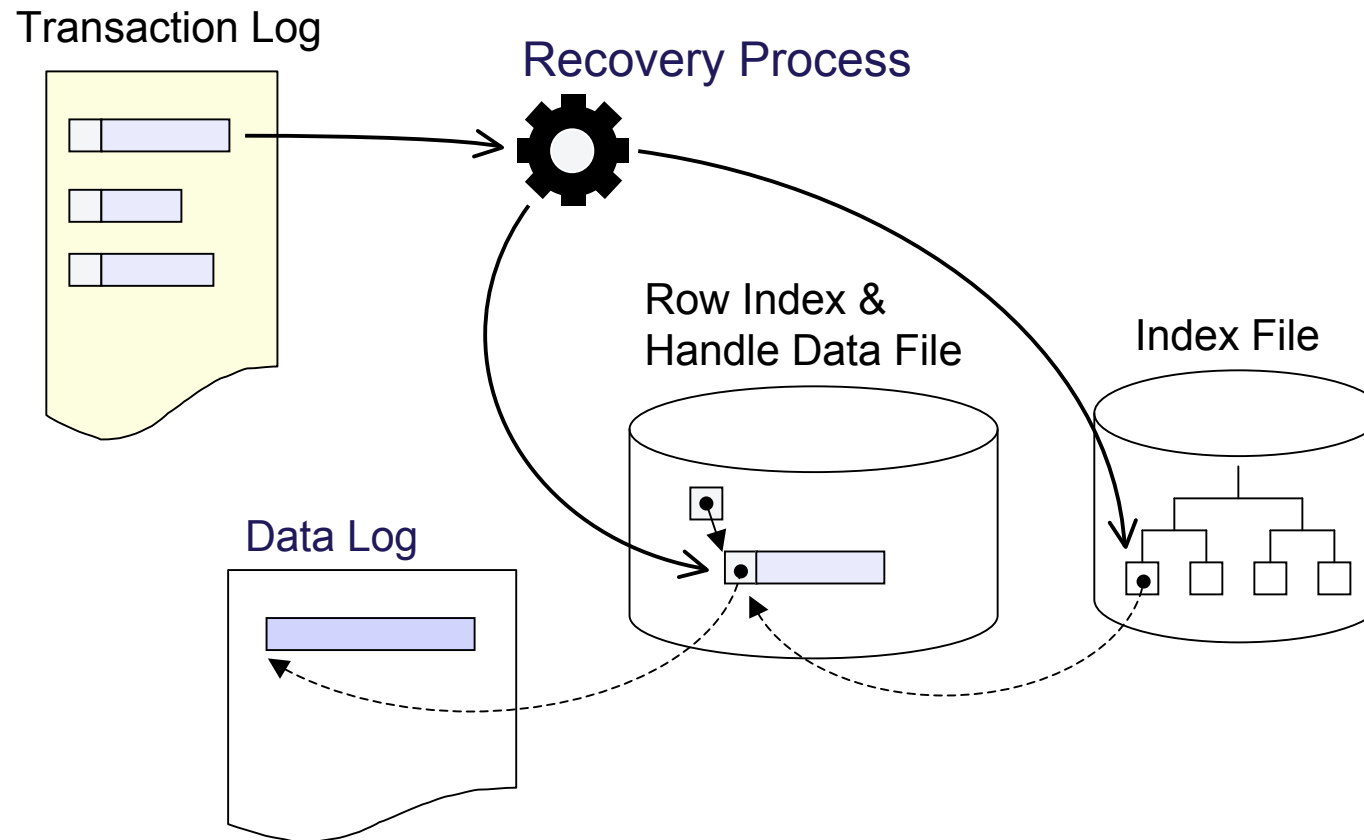


Checkpoint Thread





Recovery





Other Design Innovations

- Index entry recovery
 - Indexes need not be flushed on transaction commit.
- Operation ID's
 - Modifications normally require simultaneous update of cache and transaction log
 - Writer Thread uses the operations ID to sort changes
- Update clustering
 - New records are grouped so that they can be written together by the Writer
- Update consolidation
 - The Writer sorts updates from the log



Applications of Flexible Design

- Optimization for Solid State Drives (SSD)
- Future of PBXT: High Availability (HA) solutions.

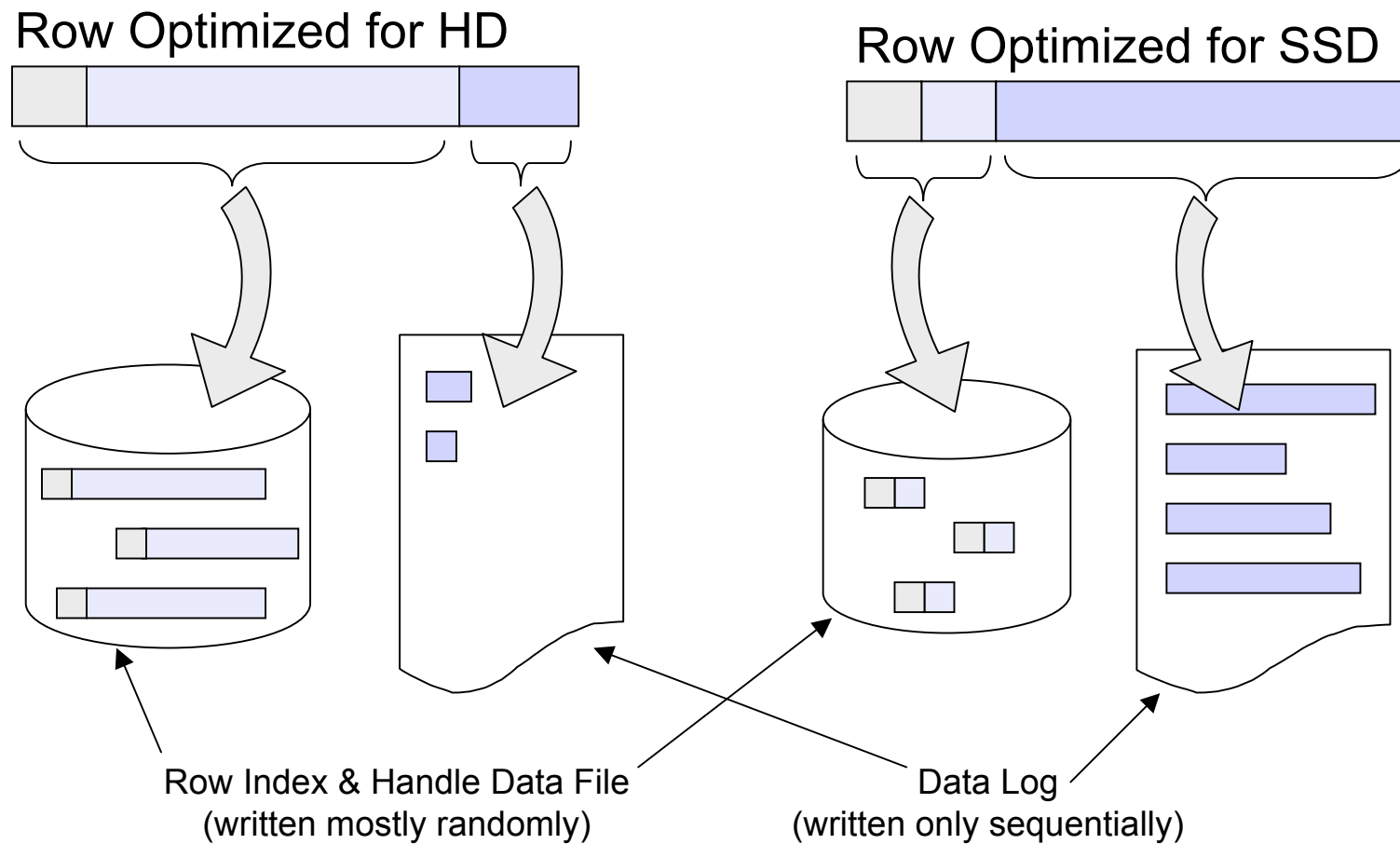


SSD Performance

	SSD	HD
Random writes	269/s	175/s
Random writes (in cache)	271/s	427,204/s
Sequential writes	112,961/s	71,975/s
Sequential writes (in cache)	559,003/s	721,709/s
Random reads	6,925/s	225/s
Random reads (from cache)	650,533/s	676,956/s

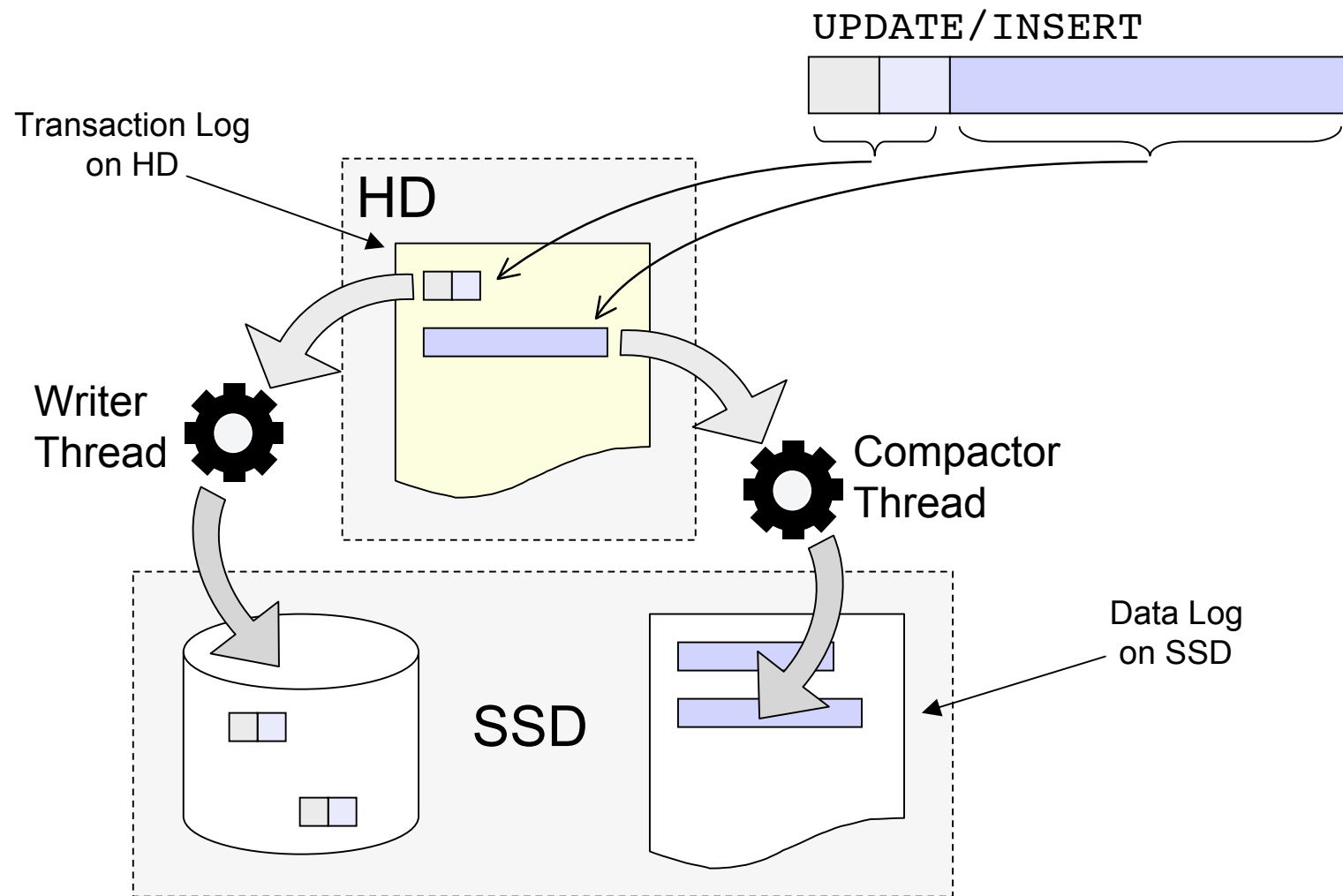


Optimizing for SSD/HD



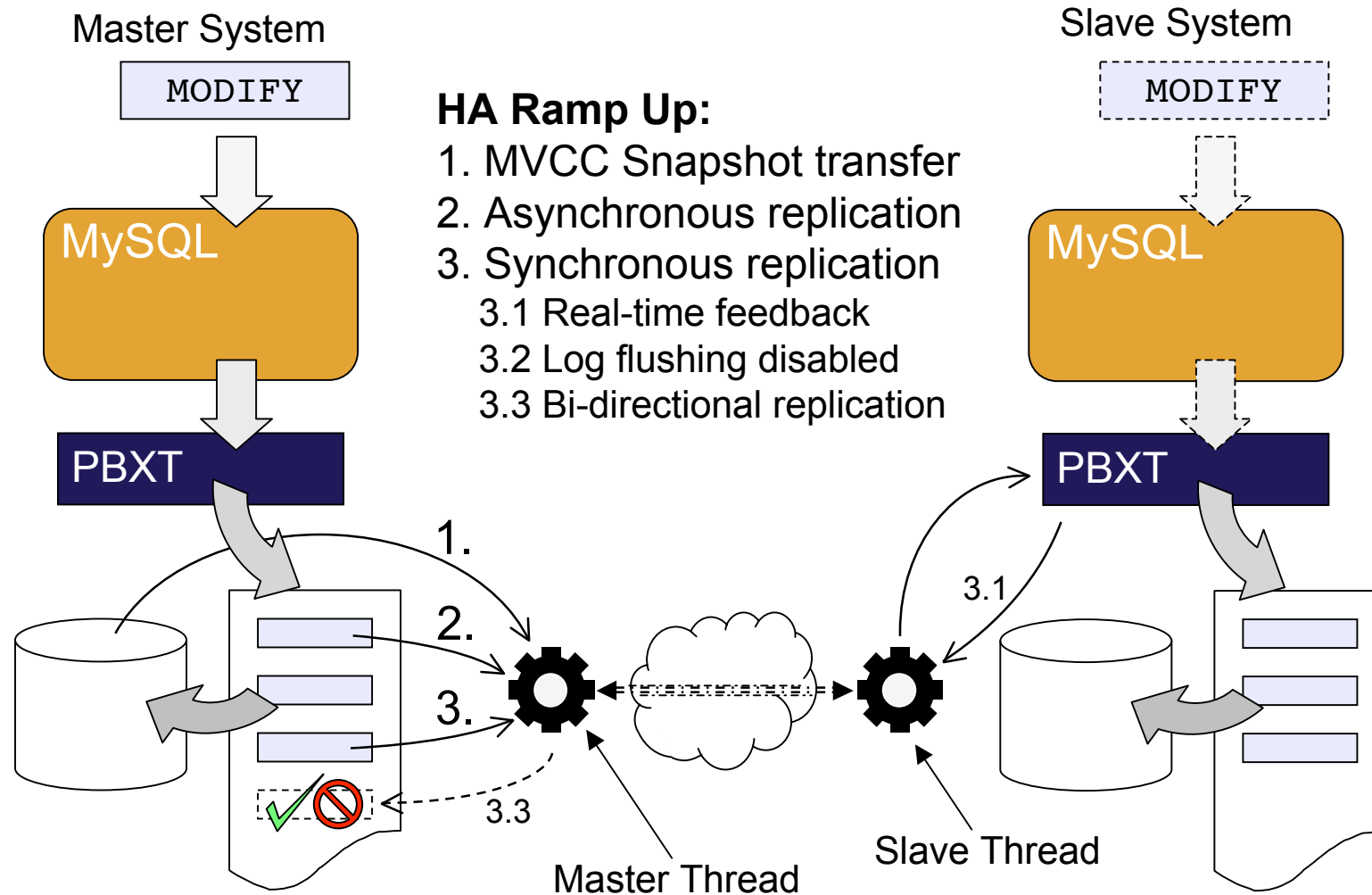


Combination HD & SSD





How HA will Work?



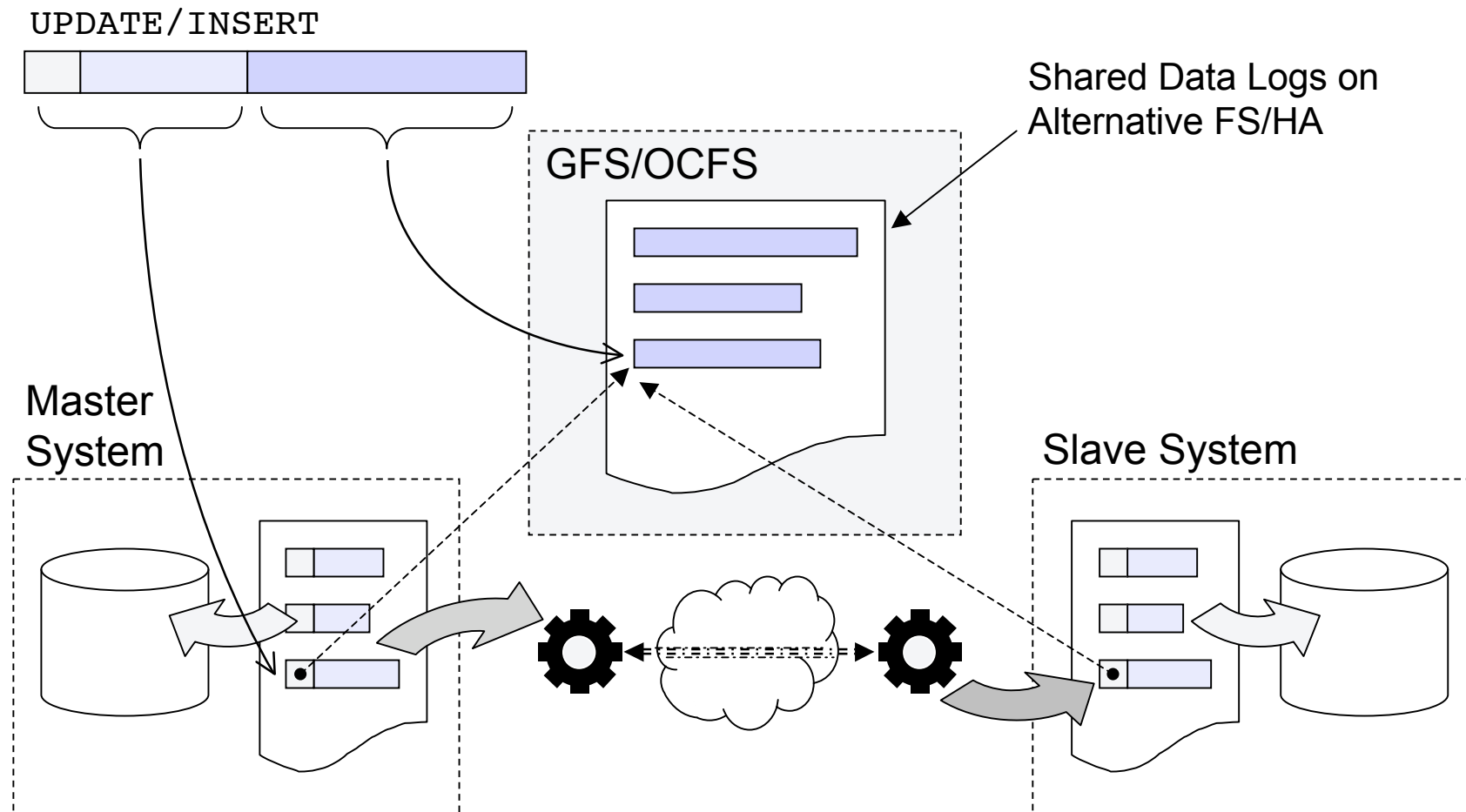


Advantages of this Solution

- Does not require writing or synchronizing with the binary log.
- Allows for rapid failover.
- Changes cannot be lost.
- Slave is online and can be used for reading or backup.
- Scalable:
 - Multiple-reader slaves
 - Master-to-master (switch master)
 - Bi-directional replication, scalable writes

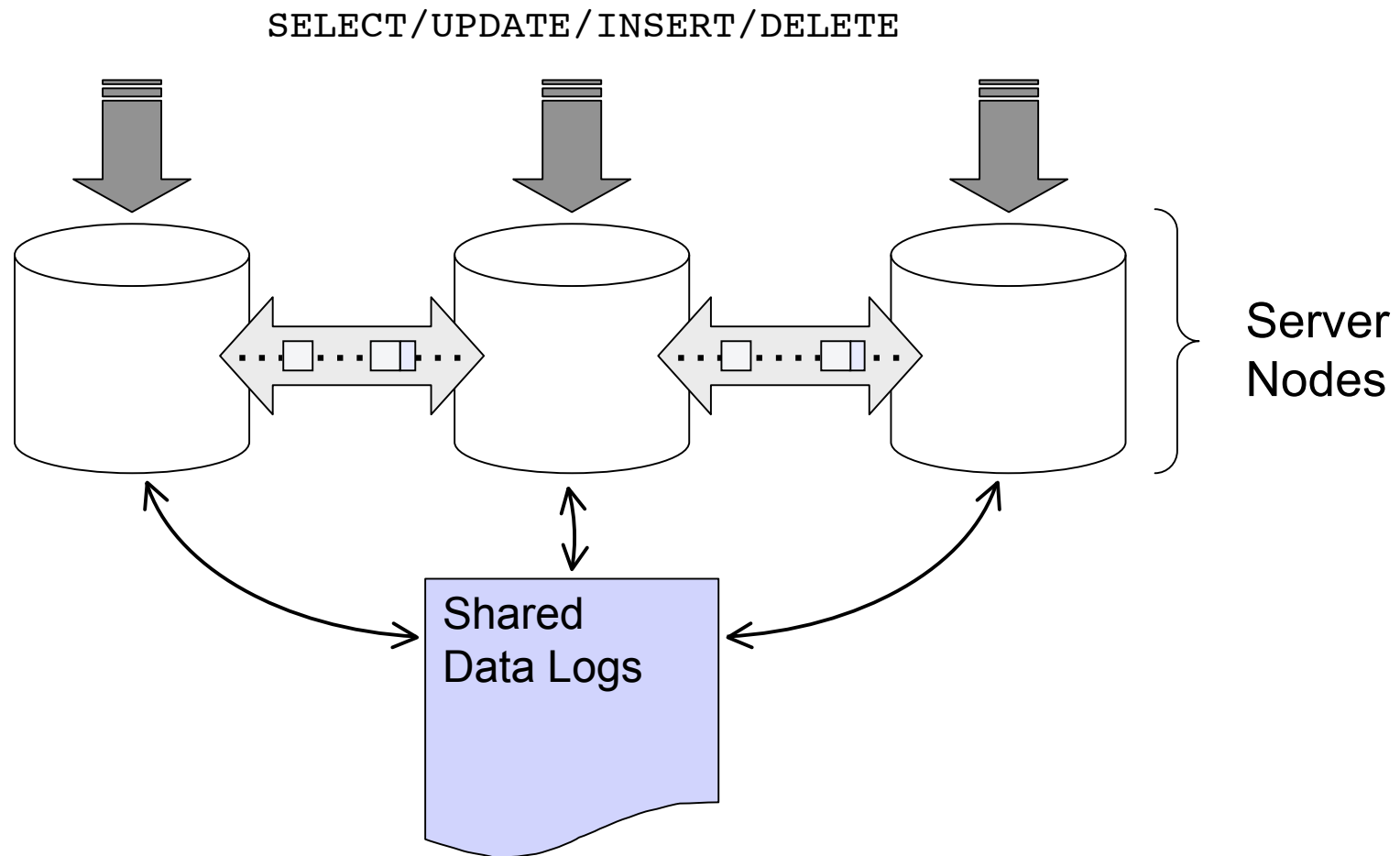


Alternative HA Configuration





Scaling Writes





PBXT Road Map

- **Q1 2008:**
 - Alpha Version
 - ACID Compliant
 - Referential Integrity
- **Q2 2008:**
 - Beta Version
 - Index Consistent Write
 - Windows Version
- **Q3 2008:**
 - Sync. HA - Alpha
- **Q4 2008:**
 - Release Candidate
- **Q1 2009:**
 - GA
 - Sync. HA - Beta



Q&A

Thanks for Listening!

<http://www.primebase.org>

<http://sourceforge.net/projects/pbxt>

<http://pbxt.blogspot.com>