Logging! (see video 14.163)
Publishing a Book

1st edition

changes:
Publishing a Book

1.1st edition

changes:

page 23:
“database” → “database”

store

log
Publishing a Book

1.2nd edition

changes:

page 23:
"database" → "database"

page 345:
"idx" → "index"
Publishing a Book

2nd edition

changes:

page 23: "database" → "database"
page 345: "idex" → "index"
page 77: "idex" → "index"
Publishing a Book

2.1st edition

changes:

page 23: "database" → "database"
page 345: "idex" → "index"
page 77: "idex" → "index"
page 75: "kamera" → "camera"
Publishing a Book

2.2nd edition

changes:

page 23:
“database” → “database”

page 345:
“idex” → “index”

page 77:
“idex” → “index”

page 75:
“kamera” → “camera”

page 143:
“big date” → “big data”
Publishing a Book

3rd edition

changes:

page 23:
“database“ → “database“

page 345:
“idex“ → “index“

page 77:
“idex“ → “index“

page 75:
“kamera“ → “camera“

page 143:
“big date“ → “big data“

new chapter on “tools“
Publishing a Book

3.1st edition

changes:

- page 23: "database" → "database"
- page 345: "idx" → "index"
- page 77: "idx" → "index"
- page 75: "kamera" → "camera"
- page 143: "big date" → "big data"
- new chapter on "tools"
- page 55: "profi" → "profile"
Logging

current edition

Read-Write File

changes:

Log File

= yet another instance of: The Data Redundancy Pattern and The All Levels are Equal Pattern
Write-Ahead Logging (WAL)

current edition

Read-Write File

changes:

Log File

2\textsuperscript{nd}: apply change to store

1\textsuperscript{st}: append to log (and flush)
Where is the Database Buffer? (see video 14.142)
Starting: No Changes yet
There was a change, let's log it.
Force the log entry to log disk.
Eventually create a new Edition 1.1
Eventually write Edition 1.1 to Disk
OR:
WAL with Multiple Storage Layers

Store

- main memory
- flash/hard disk

stable storage

- main memory
- flash/hard disk
Starting: No Changes yet

Store

main memory

flash/hard disk

stable storage

main memory

flash/hard disk
There was a change, let’s log it.
But, First: create a new Edition 1.1 in Main Memory

Store

main memory

flash/hard disk

stable storage

main memory

page 23: “database” → “database”

flash/hard disk
Then: Force the log entry to disk.
Then write Edition 1.1 to Disk

Store

main memory

flash/hard disk

stable storage

main memory

page 23: "database" → "database"

flash/hard disk

page 23: "database" → "database"
OR:
(This would be Non-WAL)
Non-WAL with Multiple Storage Layers

Store

- main memory
- flash/hard disk

stable storage

- main memory
- flash/hard disk
Starting: No Changes yet

Store

- Main memory
- Flash/hard disk

Stable storage

- Main memory
- Flash/hard disk
There was a change, let’s log it.
First: create a new Edition 1.1 in Main Memory

Store

main memory
flash/hard disk

Stable storage

main memory
page 23: "database" → "database"
flash/hard disk
Then write Edition 1.1 to Disk
Then: Force the log entry to disk.
WAL Principle

when committing a transaction:

first:     force log entry to log disk /stable store /
then:      write changed page to disk store

when writing back any dirty page to the disk store:

first:     force all corresponding log entries to log disk
then:      write changed page to disk store

overall:
1. Flush log entries
2. Flush page or disk store
Credits and Copyrights

© iStock.com:
hidesy; moenez; Rastan; hatman12; mtphoto; nickp37; voyager624

CC:
Appaloosa
http://commons.wikimedia.org/wiki/File:DRAM_DDR2_512.jpg
http://creativecommons.org/licenses/by-sa/3.0/deed.en

Lasse Fuss
http://commons.wikimedia.org/wiki/File:Lufthansa_A380_D-ALMA-1.jpg
http://creativecommons.org/licenses/by-sa/3.0/deed.en

and public domain