A Brief History

In The Beginning... (circa Y2K)

Key/Value Databases

- Client-side
- Application Scale
- Low Administration

SQL Databases

- Server-side
- Enterprise Scale
- High Administration

In The Beginning... (circa Y2K)

GDBM

KV versus SQL

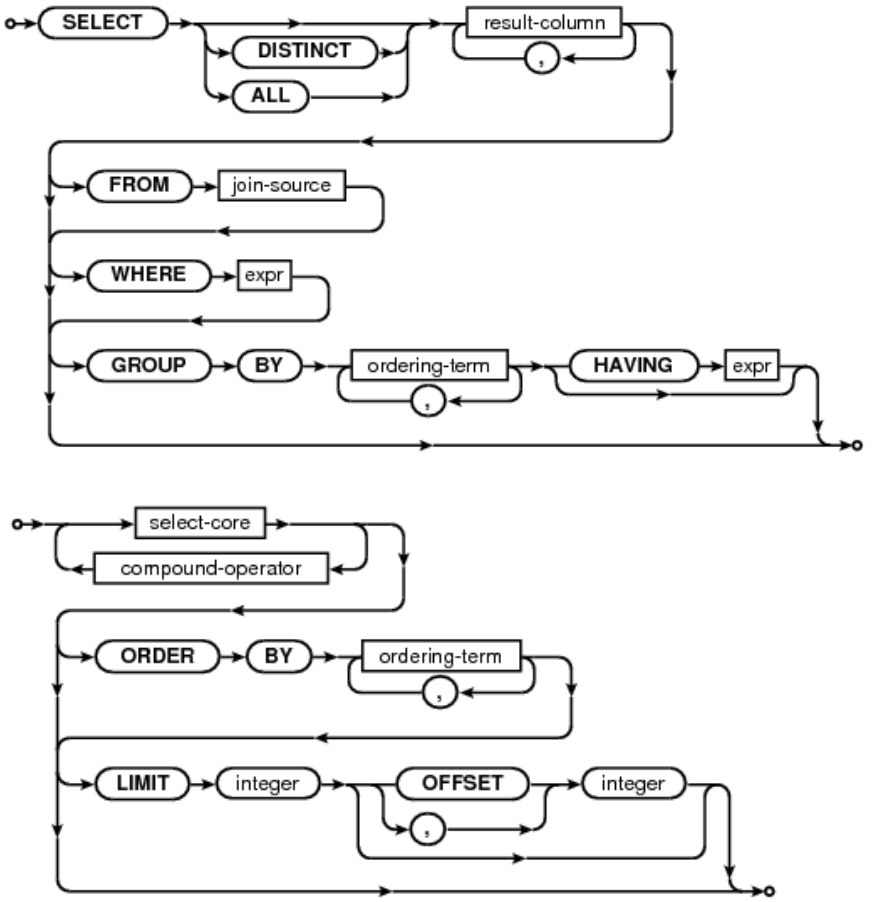
- Easy to understand
- Simple API
- Low-level operations
- Compact implementation
- Less overhead
- Faster primitives

- Steep learning curve
- Rich query language
- Promotes code with
  - Fewer errors
  - More features
- Better algorithms
- Situational awareness

KV versus SQL

PUT(*key*, *value*)

GET(*key*) → *value*



KV versus SQL

SELECT * FROM table1 WHERE pk=123

KV versus SQL

```
SELECT eqptid, enclosureid
  FROM eqpt
 WHERE typeid IN (
   SELECT typeid FROM typespec
    WHERE attrid=(
      SELECT attrid FROM attribute
       WHERE name='detect_autoactuate'
    )
    AND value=1
   INTERSECT
   SELECT typeid FROM typespec
    WHERE attrid=(
      SELECT attrid FROM attribute
       WHERE name='algorithm'
    )
    AND value IN ('sensor','wetbulb')
 )
```

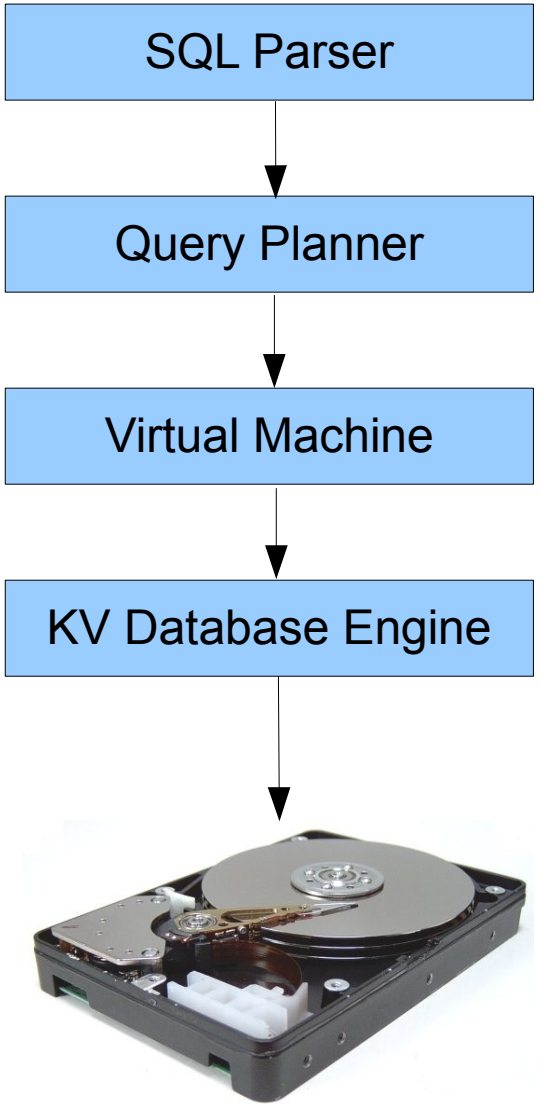
KV versus SQL

- ***How*** to compute it
- A few 1000 lines
- Low-level
- Bare metal
- Many bugs
- Heads down

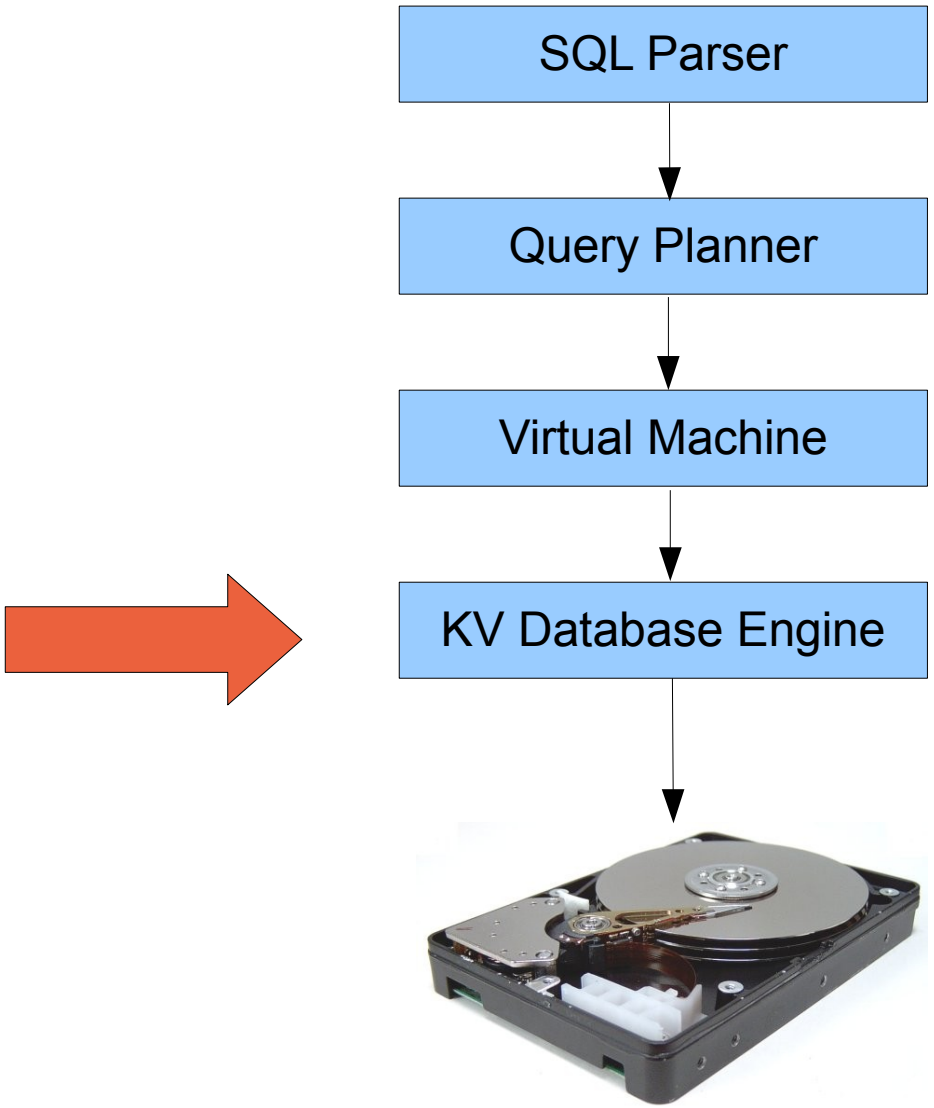
- ***What*** to compute
- A few lines of code
- High-level
- Abstract
- Fewer bugs
- Heads up

SQL promotes situational awareness

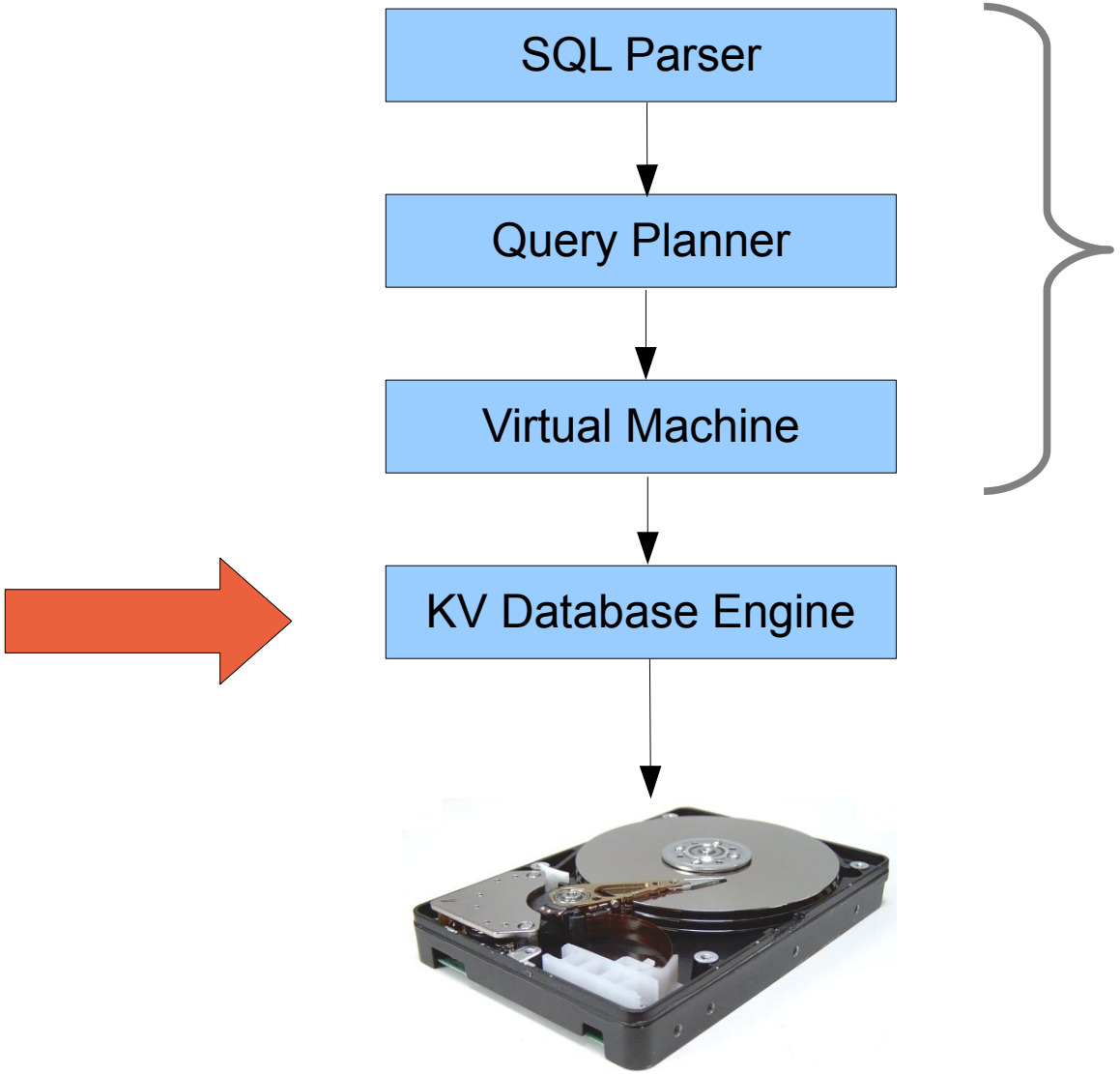
KV versus SQL

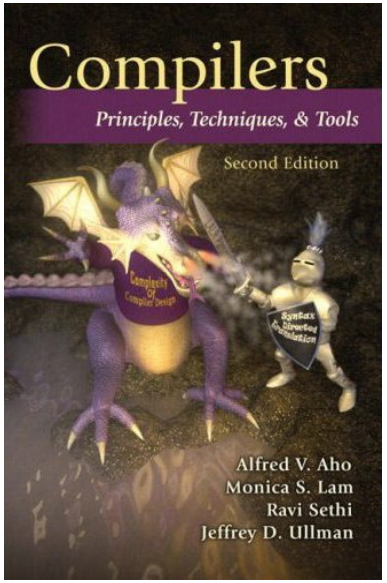


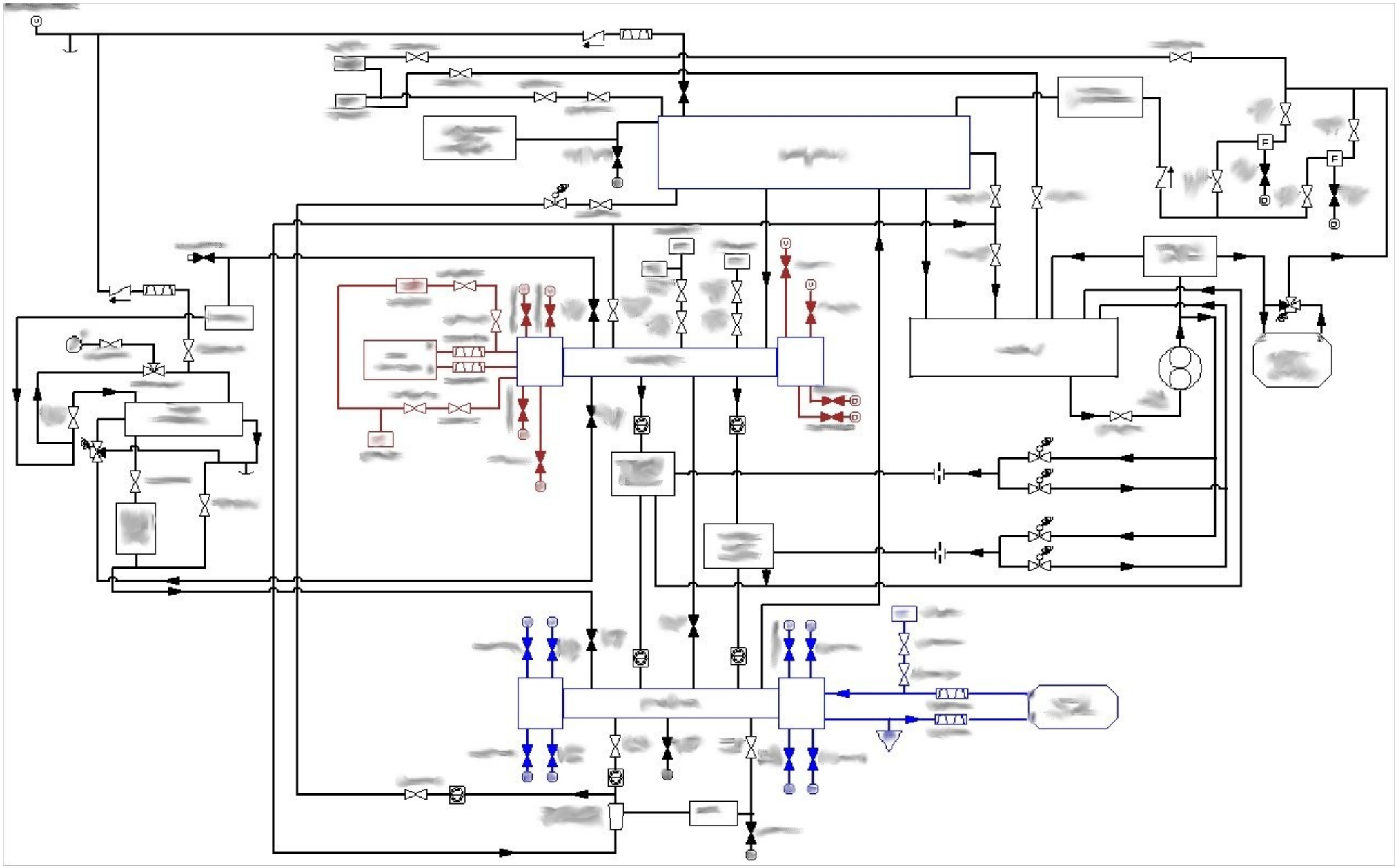
KV versus SQL

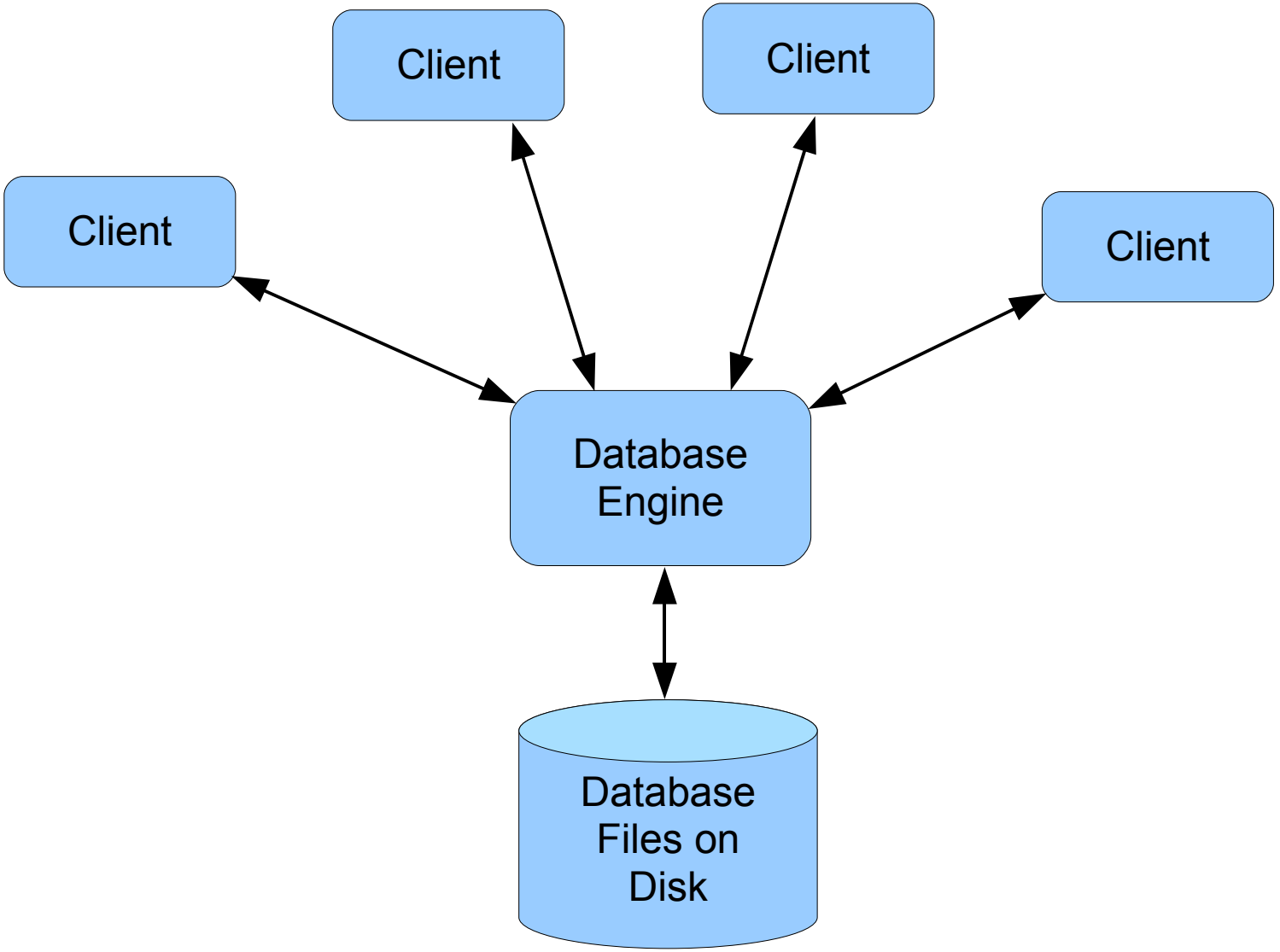


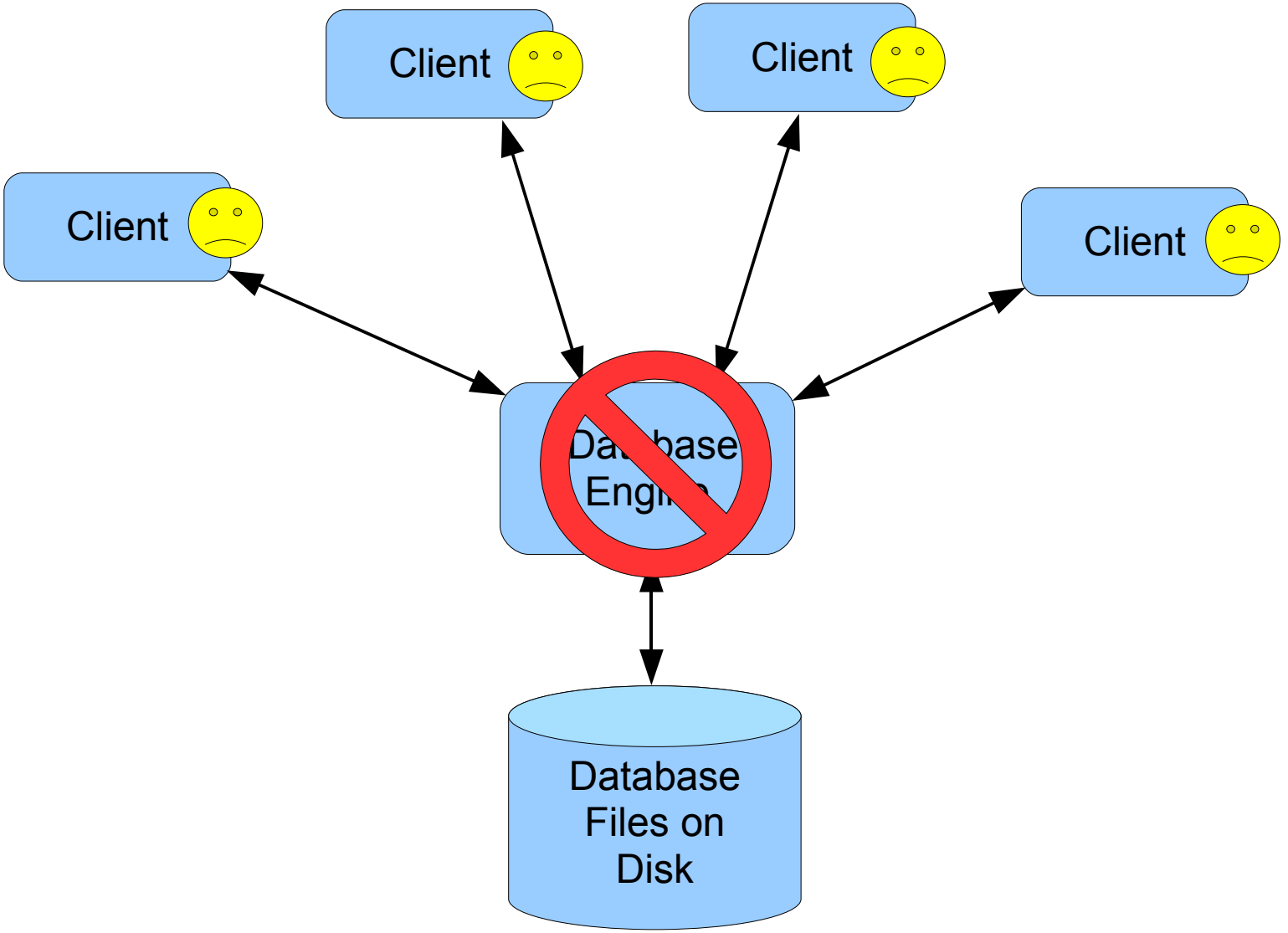
KV versus SQL

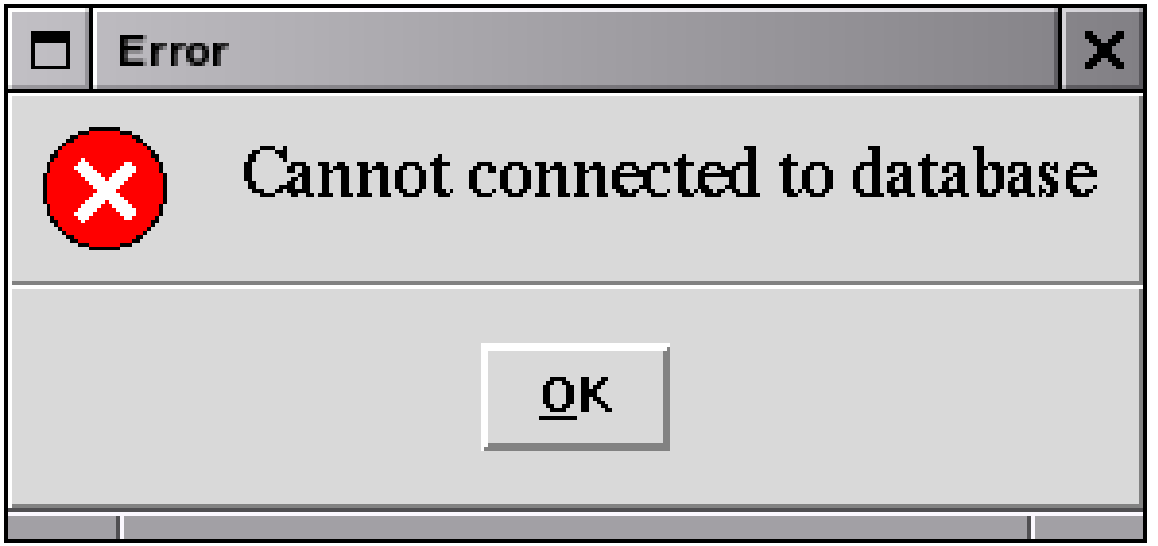


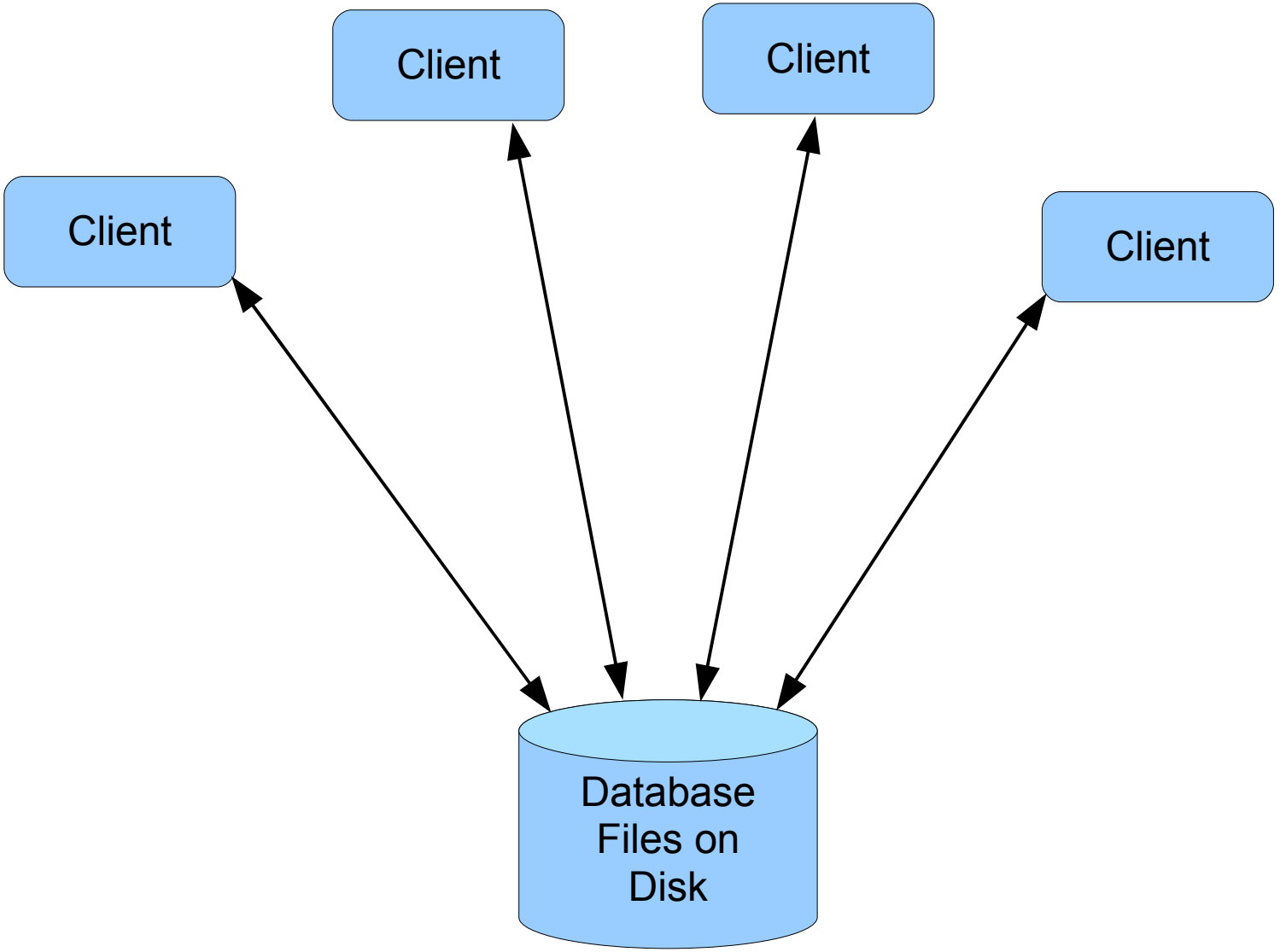





Client
Client
Client
Client
Client
Database
Engine
Database
Files on
Disk


Client
Client
Client
Client
Database
Files on
Disk


Error
Cannot connected to database
OK


Client
Client
Client
Client
Database
Files on
Disk

SQLite 1.0

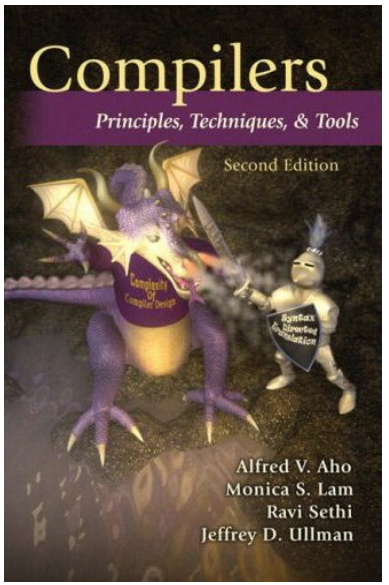
GDBM

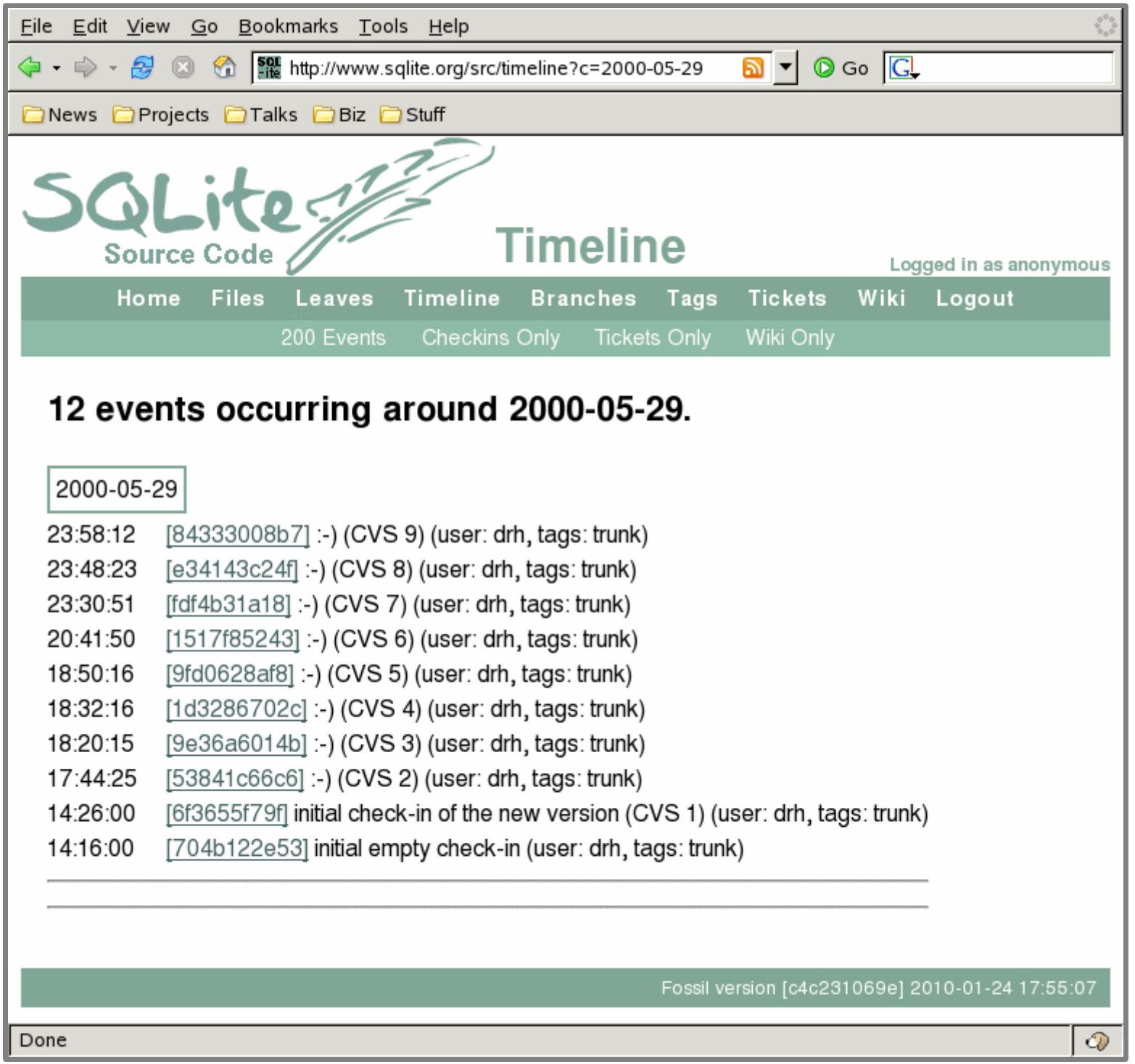
SQL Parser

Query Planner

Virtual Machine

KV Database Engine



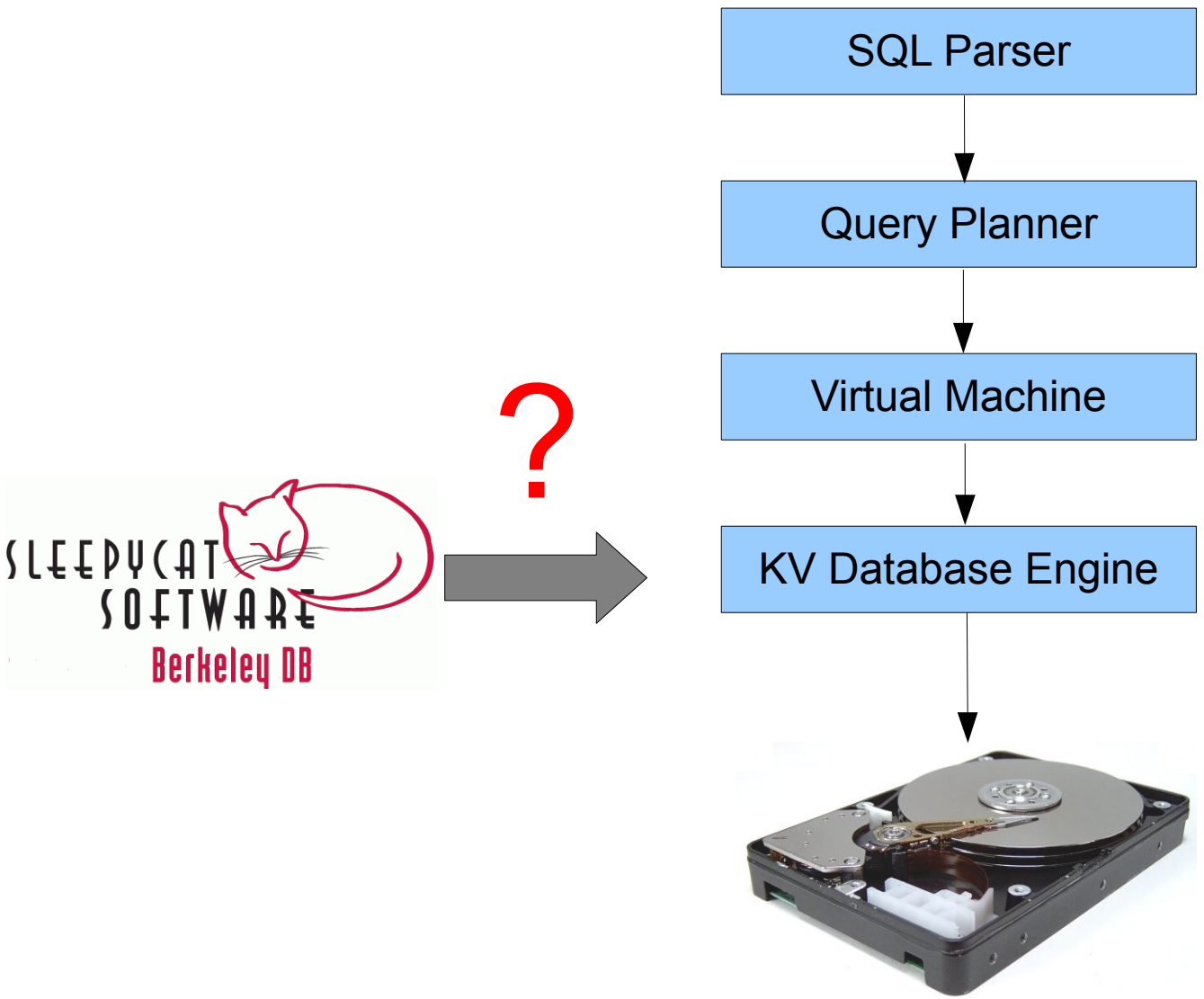

File Edit View Go Bookmarks Tools Help
http://www.sqlite.org/src/timeline?c=2000-05-29
Go
News Projects Talks Biz Stuff
SQLite
Source Code
Timeline
Logged in as anonymous
Home Files Leaves Timeline Branches Tags Tickets Wiki Logout
200 Events Checkins Only Tickets Only Wiki Only
12 events occurring around 2000-05-29.
2000-05-29
23:58:12 [84333008b7] :-) (CVS 9) (user: drh, tags: trunk)
23:48:23 [e34143c24f] :-) (CVS 8) (user: drh, tags: trunk)
23:30:51 [fdf4b31a18] :-) (CVS 7) (user: drh, tags: trunk)
20:41:50 [1517f85243] :-) (CVS 6) (user: drh, tags: trunk)
18:50:16 [9fd0628af8] :-) (CVS 5) (user: drh, tags: trunk)
18:32:16 [1d3286702c] :-) (CVS 4) (user: drh, tags: trunk)
18:20:15 [9e36a6014b] :-) (CVS 3) (user: drh, tags: trunk)
17:44:25 [53841c66c6] :-) (CVS 2) (user: drh, tags: trunk)
14:26:00 [6f3655f79f] initial check-in of the new version (CVS 1) (user: drh, tags: trunk)
14:16:00 [704b122e53] initial empty check-in (user: drh, tags: trunk)
Fossil version [c4c231069e] 2010-01-24 17:55:07
Done


File Edit View Go Bookmarks Tools Help
http://www.sqlite.org/src/timeline?c=2000-08-17+10
Go
News Projects Talks Biz Stuff
SQLite
Source Code
Timeline
Logged in as anonymous
Home Files Leaves Timeline Branches Tags Tickets Wiki Logout
200 Events Checkins Only Tickets Only Wiki Only
10 events occurring around 2000-08-17 10:25:00.
2000-08-18
10:00:00 [e8521fc10d] Version 1.0.1 (CVS 498) (user: drh, tags: trunk)
09:58:52 [0a0576e2f9] :-) (CVS 135) (user: drh, tags: trunk)
09:34:19 [862b649204] configure script bug (CVS 134) (user: drh, tags: trunk)
09:33:40 [c773a449b1] configure script bug (CVS 133) (user: drh, tags: trunk)
2000-08-17
10:25:00 [f37dd18e3f] Version 1.0 (CVS 499) (user: drh, tags: trunk)
10:22:34 [5ec2b09478] add version numbering (CVS 132) (user: drh, tags: trunk)
09:50:00 [897b4bc0e9] allow readonly access when write permission denied (CVS 131) (user: drh, tags: trunk)
2000-08-09
17:17:25 [e8882dac23] bug fix (CVS 130) (user: drh, tags: trunk)
Done

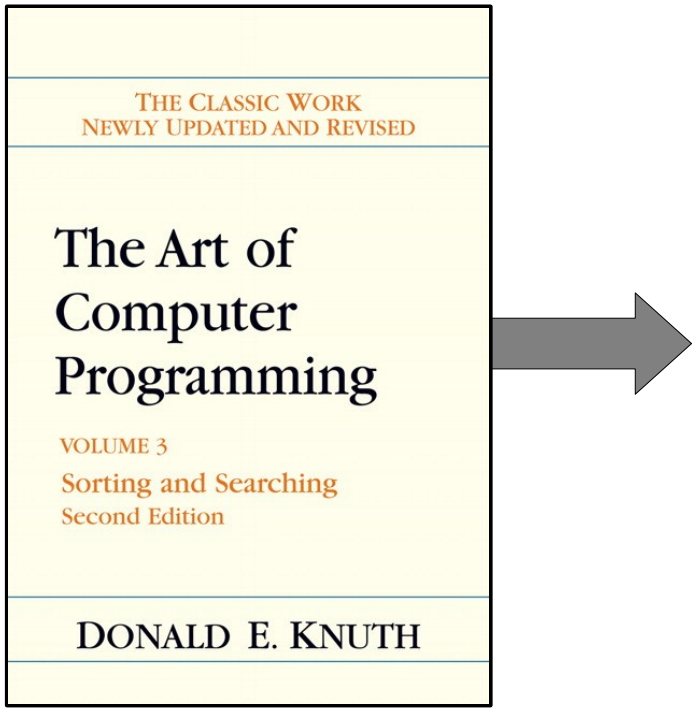
Issues with GDBM

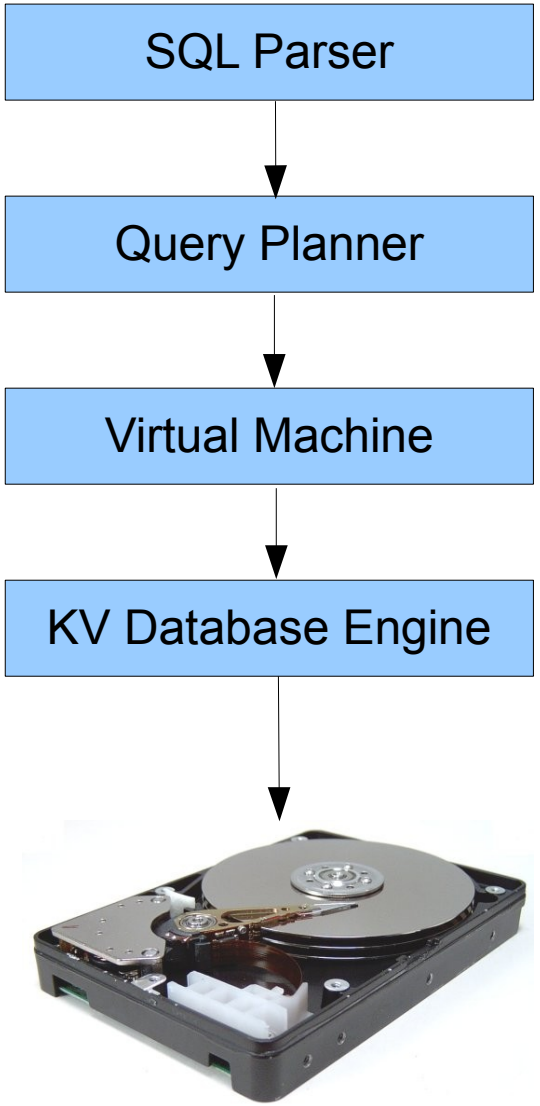
- Separate file for each table and index
- Hash based – cannot do range queries
- No transactions
  - Updates are not atomic
  - Cannot rollback
- File corruption on crash or power loss
- GPL

Change The Underlying KV Database?

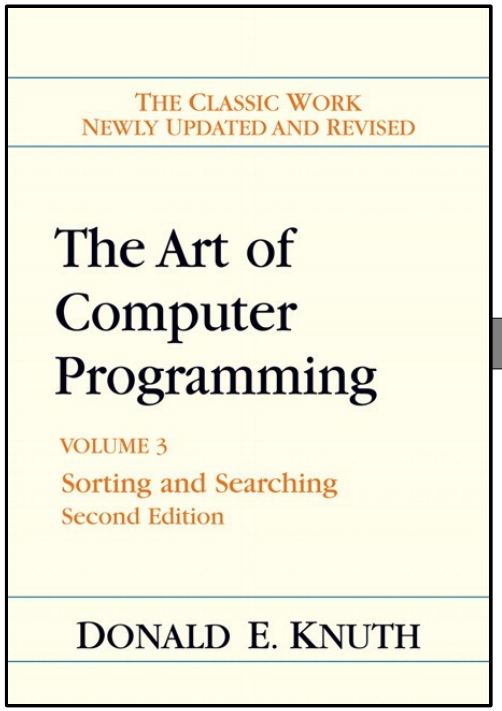


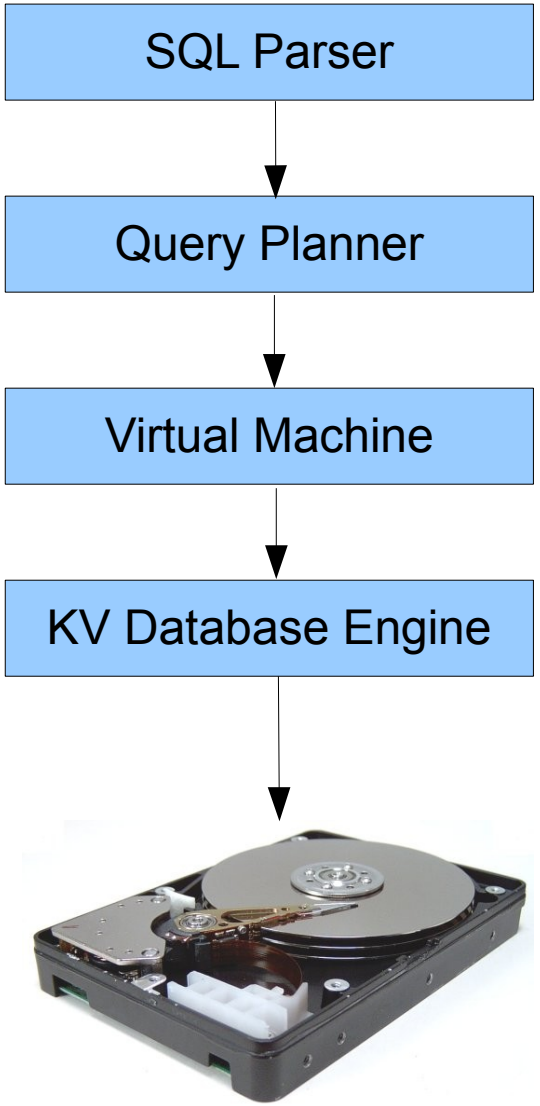
SQLite 2.0


THE CLASSIC WORK
NEWLY UPDATED AND REVISED
The Art of Computer Programming
VOLUME 3
Sorting and Searching
Second Edition
DONALD E. KNUTH


SQL Parser
Query Planner
Virtual Machine
KV Database Engine

SQLite 2.0


THE CLASSIC WORK
NEWLY UPDATED AND REVISED
The Art of Computer Programming
VOLUME 3
Sorting and Searching
Second Edition
DONALD E. KNUTH


SQL Parser
Query Planner
Virtual Machine
KV Database Engine


File Edit View Go Bookmarks Tools Help
http://www.sqlite.org/src/timeline?c=2001-09-28+23
Go
News Projects Talks Biz Stuff
SQLite
Source Code
Timeline
Logged in as anonymous
Home Files Leaves Timeline Branches Tags Tickets Wiki Logout
200 Events Checkins Only Tickets Only Wiki Only
9 events occurring around 2001-09-28 23:15:00.
2001-10-02
13:05:00 [e498084940] Version 2.0.1 (CVS 469) (user: drh, tags: trunk)
13:01:49 [4b7710e2da] Remove C++ comments from btree.c. (CVS 277) (user: drh, tags: trunk)
2001-10-01
14:29:23 [20382325c7] The .dump output uses INSERT instead of COPY now. Expression syntax of the form "expr NOT NULL" is now supported. (CVS 276) (user: drh, tags: trunk)
2001-09-28
23:15:00 [c0a8a1fb42] Version 2.0.0 (CVS 470) (user: drh, tags: trunk)
23:11:24 [4b4bfc6290] Documentation updates. (CVS 275) (user: drh, tags: trunk)
18:14:17 [0ffab36d1f] Remove reference to GDBM in the documentation of the "sqlite" command-line utility. (CVS 274) (user: drh, tags: trunk)
18:10:56 [7e79e91b03] Line tclsqlite.so against the stub library. (CVS 273) (user: drh, tags: trunk)
Done

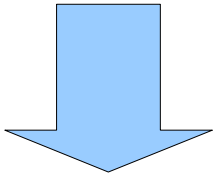
SQLite 2.X

- Single-file database
- Useful subset of SQL
- Everything is a string
- Transactions with atomic commit & rollback
- Public domain
- About 250KiB size

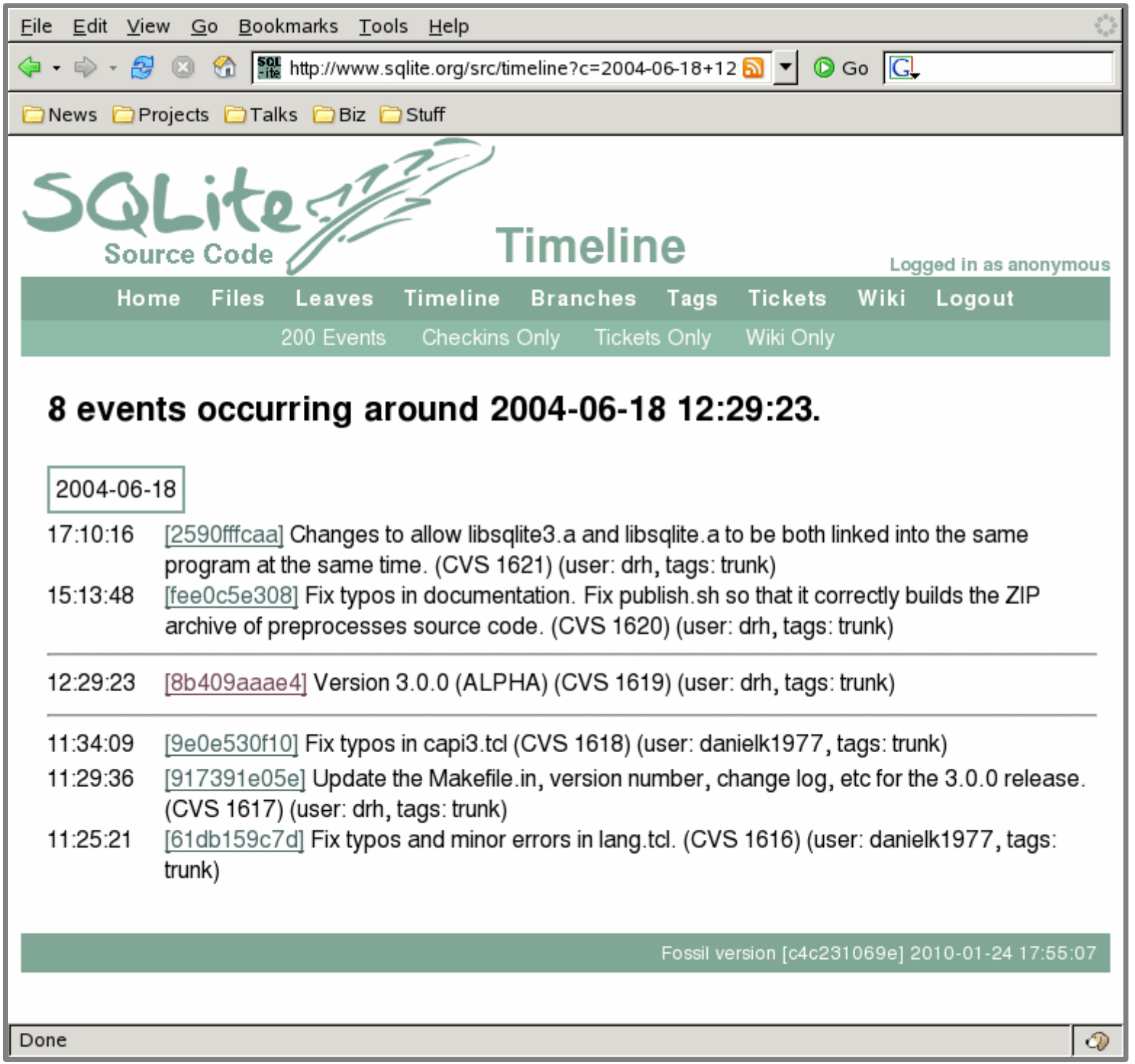
2001-11-12	2.1.0
2001-12-22	2.2.0
2002-01-30	2.3.0
2002-03-10	2.4.0
2002-06-17	2.5.0
2002-07-17	2.6.0
2002-08-25	2.7.0
2003-02-16	2.8.0

(early 2004:) AOL needed....

- UTF-16 support
- BLOB support
- Collating sequences
- Two-phase commit of ATTACH-ed databases



SQLite 3.0.0


File Edit View Go Bookmarks Tools Help
http://www.sqlite.org/src/timeline?c=2004-06-18+12
Go
News Projects Talks Biz Stuff
SQLite
Source Code
Timeline
Logged in as anonymous
Home Files Leaves Timeline Branches Tags Tickets Wiki Logout
200 Events Checkins Only Tickets Only Wiki Only
8 events occurring around 2004-06-18 12:29:23.
2004-06-18
17:10:16 [2590fffcaa] Changes to allow libsqlite3.a and libsqlite.a to be both linked into the same program at the same time. (CVS 1621) (user: drh, tags: trunk)
15:13:48 [fee0c5e308] Fix typos in documentation. Fix publish.sh so that it correctly builds the ZIP archive of preprocesses source code. (CVS 1620) (user: drh, tags: trunk)
12:29:23 [8b409aaae4] Version 3.0.0 (ALPHA) (CVS 1619) (user: drh, tags: trunk)
11:34:09 [9e0e530f10] Fix typos in capi3.tcl (CVS 1618) (user: danielk1977, tags: trunk)
11:29:36 [917391e05e] Update the Makefile.in, version number, change log, etc for the 3.0.0 release. (CVS 1617) (user: drh, tags: trunk)
11:25:21 [61db159c7d] Fix typos and minor errors in lang.tcl. (CVS 1616) (user: danielk1977, tags: trunk)
Fossil version [c4c231069e] 2010-01-24 17:55:07
Done

2004-06-18	3.0.0
2005-01-21	3.1.0
2005-03-21	3.2.0
2006-01-10	3.3.0
2007-06-18	3.4.0
2007-09-04	3.5.0
2008-07-16	3.6.0

Currently at 3.6.22

- Almost All SQL
  - Missing RIGHT and FULL OUTER JOIN
  - Incomplete ALTER TABLE
- Faster than ever
  - Improved query planner
  - Less I/O
- 100% branch testing
- FOREIGN KEY
- SAVEPOINT
- Full-text search
- R-Tree indexes
- 282 KiB binary
- Terabyte Databases

Other Features Of SQLite

- App-defined functions
- App-defined collating sequences
- UTF8 or UTF16
- Robust against power loss
- Robust against malloc() failures
- Live backup
- Virtual Tables
- ATTACH DATABASE
- Gigibyte size BLOBs and strings
- Robust against I/O errors
- Zero-malloc option

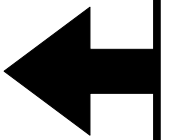
SQLite bridges the gap...

- Like BerkeleyDB and GDBM:
  - Direct to disk; serverless; client-side
  - Linked into application
  - Zero administration
- Like PostgreSQL, Oracle, etc:
  - High-level query language
  - Transactional & Relational
  - Heads-up programming; situational awareness

In The Beginning... (circa Y2K)

GDBM

... Squeezing out BDB and GDBM



Other client-side SQL engines

One new notable KV database...

Do not confuse SQLite with server database engines....


iPod
Music
Photos
Extras
Settings
Shuffle Songs
MENU

ORACLE®

is to

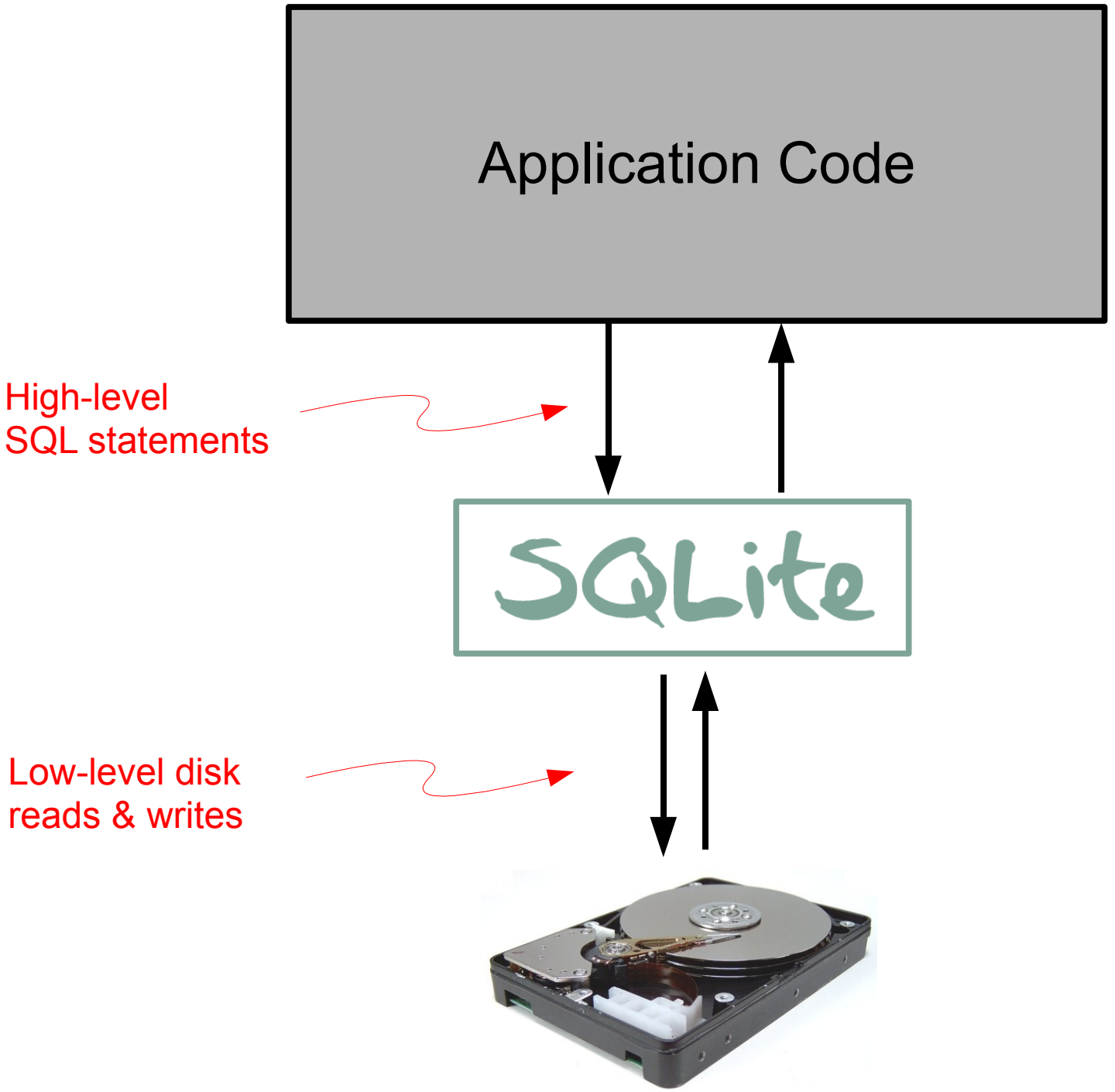
SQLite

as

is to

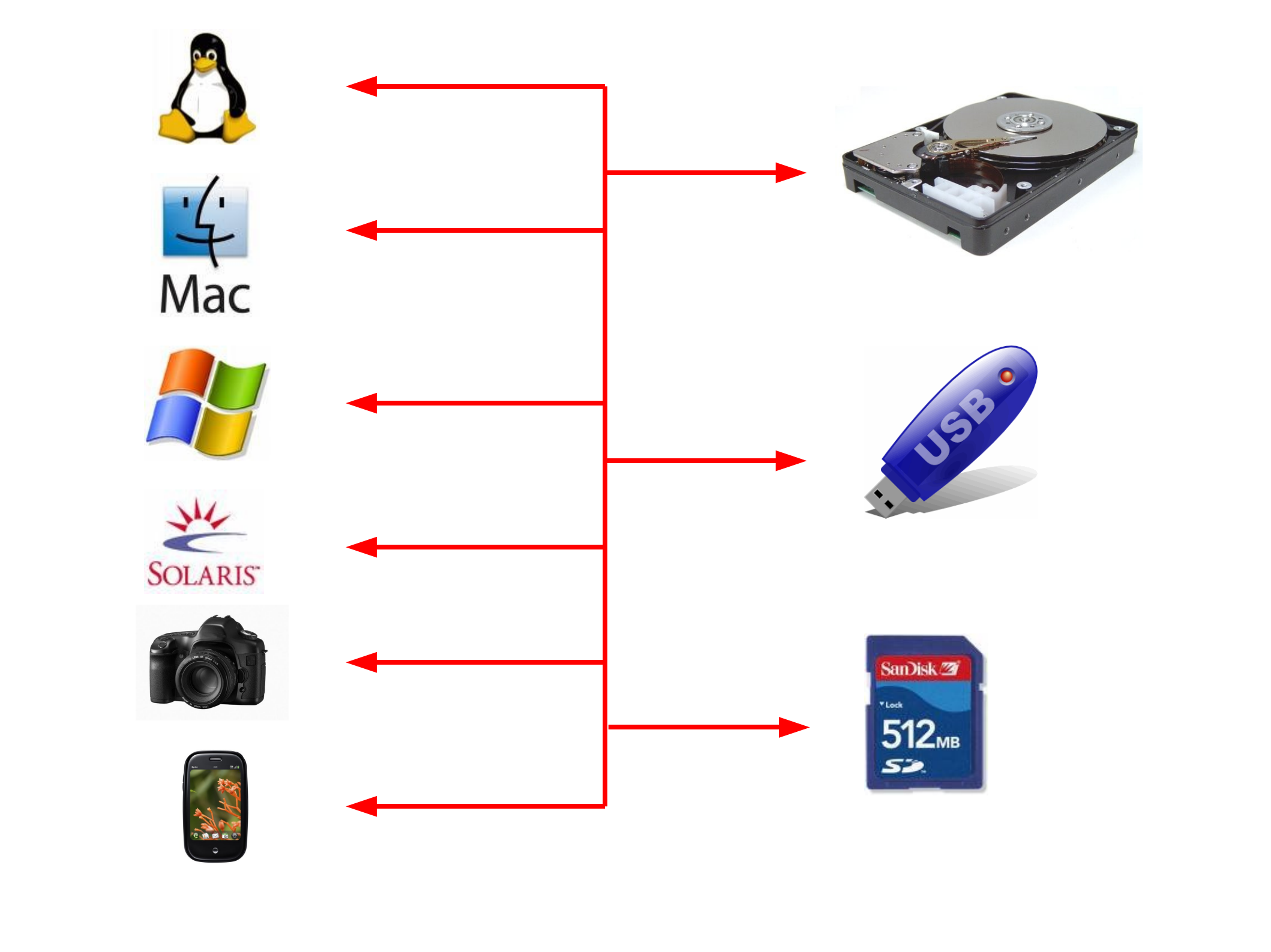
- SQLite does not compete with Oracle

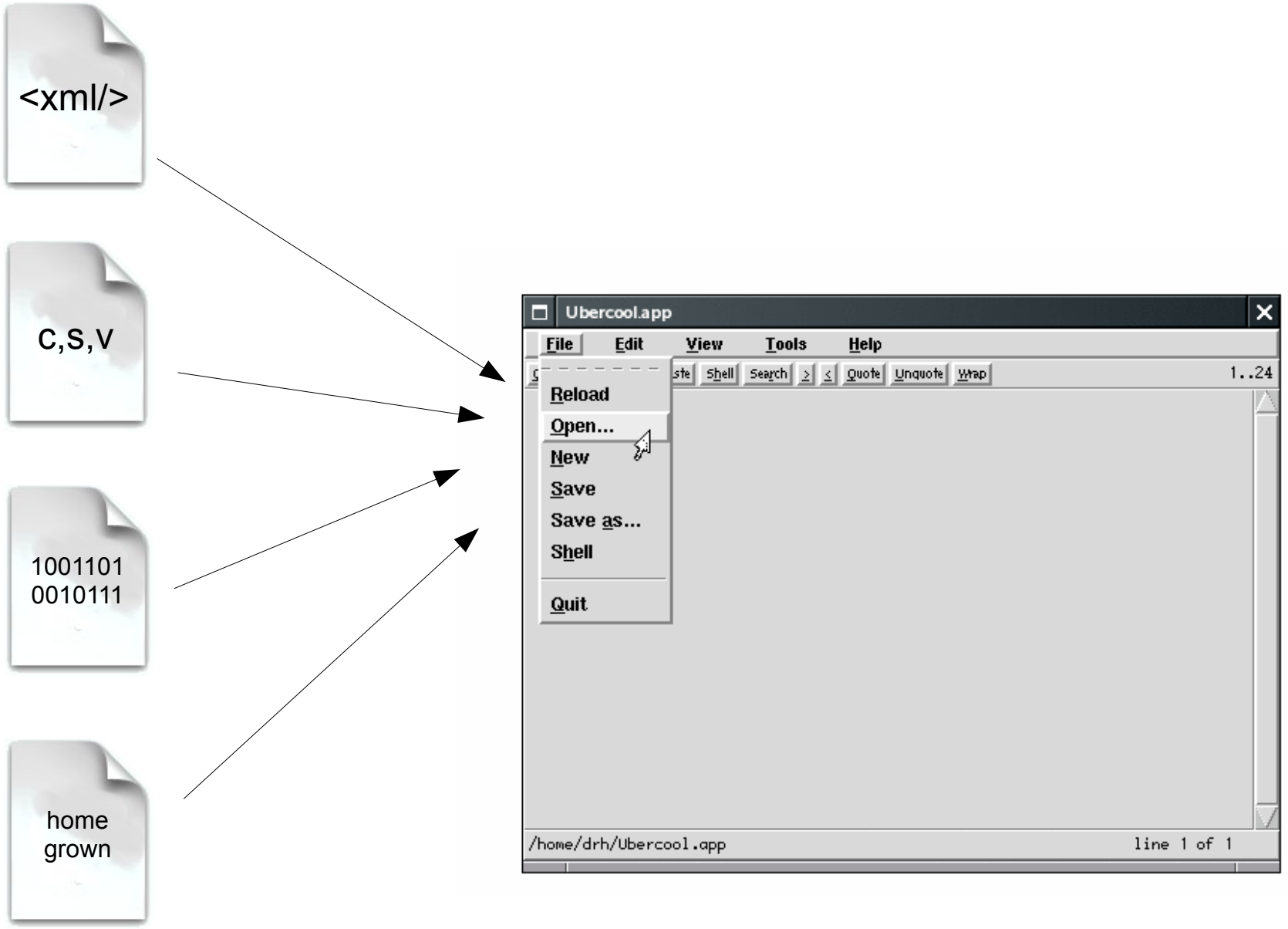
- SQLite does not compete with Oracle
- SQLite competes with **fopen()**

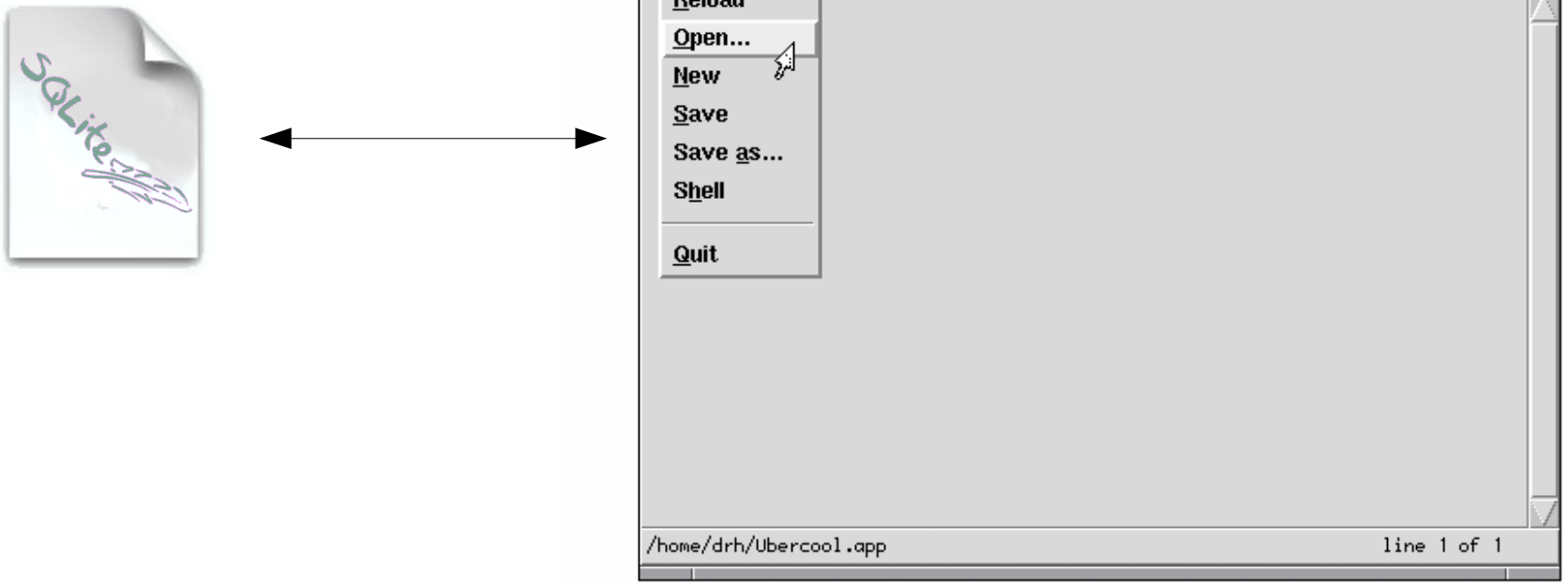

Application Code
High-level
SQL statements
SQLite
Low-level disk
reads & writes

Portable File Format

- A database is a single ordinary disk file
- No special naming conventions or required file suffixes
- Cross-platform: big/little-endian and 32/64-bit
- Backwards compatible through 3.0.0
- Promise to keep it compatible moving forward
- Not tied to any particular programming language.


Mac
SOLARIS
USB
SanDisk
512MB
SD


<xml/>
C,S,V
1001101
0010111
home
grown
Ubercool.app
File
Edit
View
Tools
Help
Shell
Search
Quote
Unquote
Wrap
1..24
Reload
Open...
New
Save
Save as...
Shell
Quit
/home/drh/Ubercool.app
line 1 of 1


SQLite
Ubercool.app
File
Edit
View
Tools
Help
Shell
Search
Quote
Unquote
Wrap
1..24
Reload
Open...
New
Save
Save as...
Shell
Quit
/home/drh/Ubercool.app
line 1 of 1

SQLite as file format freebies

- No parsing and generating code to write
- Atomic updates
- Fast, built-in searching
- Access via third-party tools
- Simplified upgrade migration
- Cross-platform file format
- High-level query language
- sqlite3_trace()

Small Footprint

gcc -Os -DSQLITE_THREADSAFE=0

282 KiB

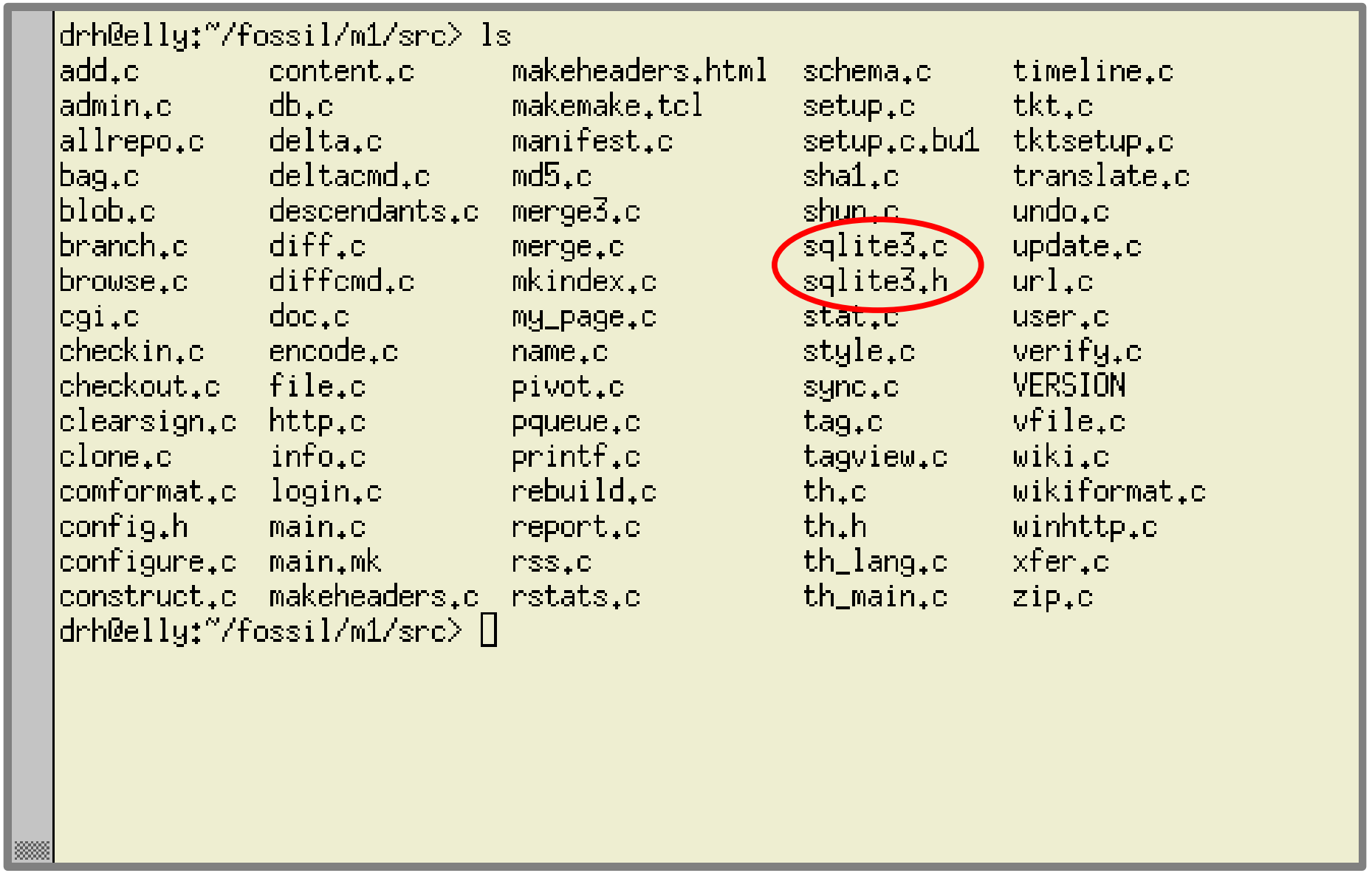
gcc -O3 -DSQLITE_ENABLE_FTS3=1 -DSQLITE_ENABLE_RTREE=1

868 KiB

As of 2010-01-25

Single Source Code File

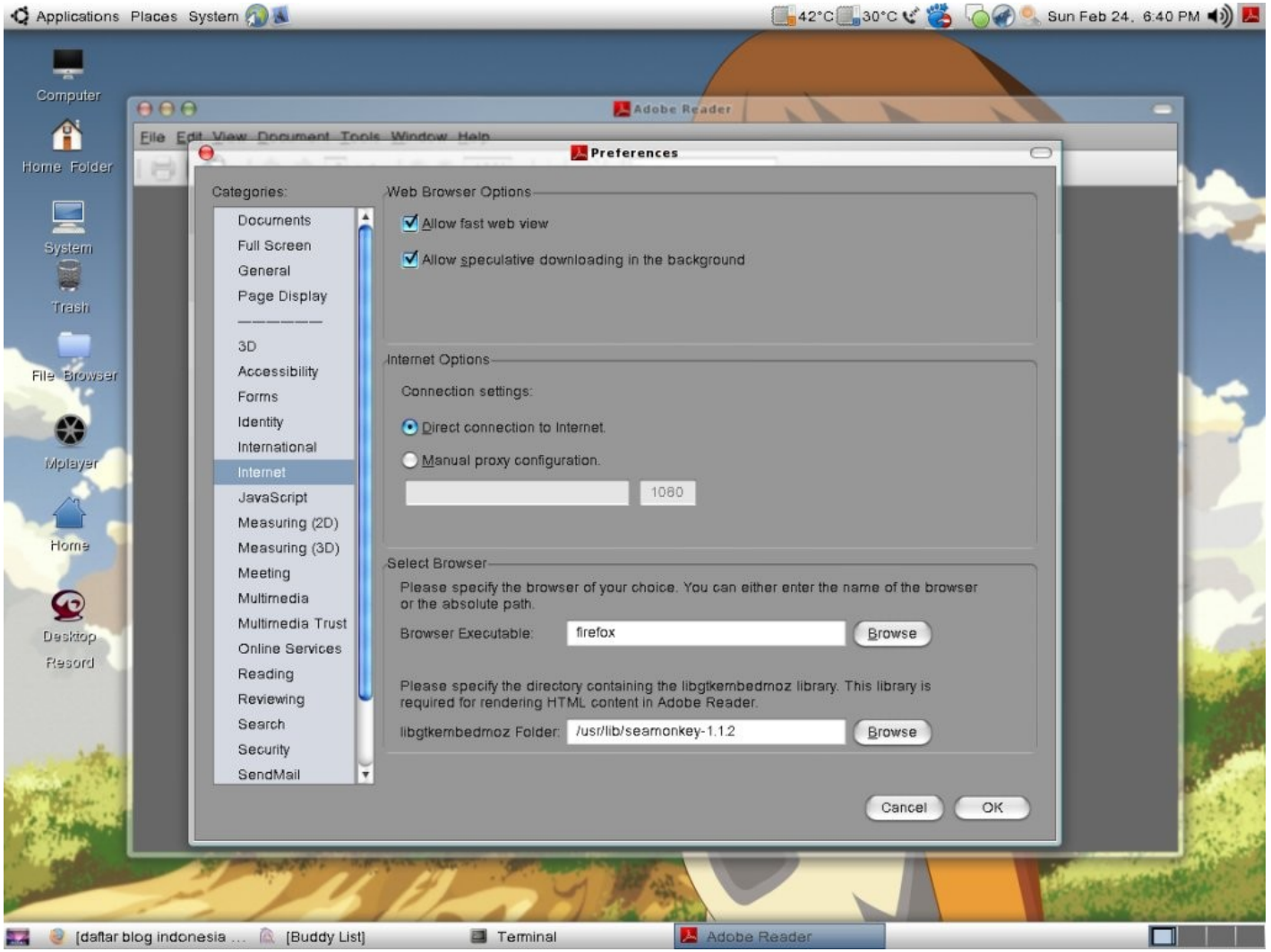
- The “amalgamation” source code file: **sqlite3.c**
- About 66,000 lines of ANSI C code
- 3.9 MB
- No other library dependencies on than standard library routines:
  - memcpy(), memset(), malloc(), free(), etc
- Very simple to add to a larger C program


drh@elly:~/fossil/m1/src> ls
add.c content.c makeheaders.html schema.c timeline.c
admin.c db.c makemake.tcl setup.c tkt.c
allrepo.c delta.c manifest.c setup.c.bu1 tktsetup.c
bag.c deltacmd.c md5.c sha1.c translate.c
blob.c descendants.c merge3.c shun.c undo.c
branch.c diff.c merge.c sqlite3.c update.c
browse.c diffcmd.c mkindex.c sqlite3.h url.c
cgi.c doc.c my_page.c stat.c user.c
checkin.c encode.c name.c style.c verify.c
checkout.c file.c pivot.c sync.c VERSION
clearsign.c http.c pqueue.c tag.c vfile.c
clone.c info.c printf.c tagview.c wiki.c
comformat.c login.c rebuild.c th.c wikiformat.c
config.h main.c report.c th.h winhttp.c
configure.c main.mk rss.c th_lang.c xfer.c
construct.c makeheaders.c rstats.c th_main.c zip.c
drh@elly:~/fossil/m1/src>

Many companies and organizations use SQLite...

Adobe Photoshop Lightroom

Adobe Reader



Mozilla Firefox

symbian

Android

iPhone


Pizza
Frank's Pizza
http://www.nytimes.com
The New York Times
AT&T
9:42 AM
John Appleseed
calling...
End Call

iPod & iTunes


iTunes
Albums
Artists
Genres
Composers
Music
Movies
TV Shows
Podcasts
Applications
Radio
iTunes Store
Purchased
Steve's iPhone
Party Shuffle
Both Sides of the Gun
Ben Harper
Coco
Colbie Caillat
Datarock
Detours
Sheryl Crow
Eyes – Single
Rogue Wave
Flavors of Entanglement
Alanis Morissette
Fly Between Falls
ALO (Animal Liberation...
Gentlemen
The Afghan Whips
BECK
MODERN GUILT
Sleep Through the Static
Jack Johnson
Superhero Brother
G. Love & Special Sauce
DATAROCK

iStuff

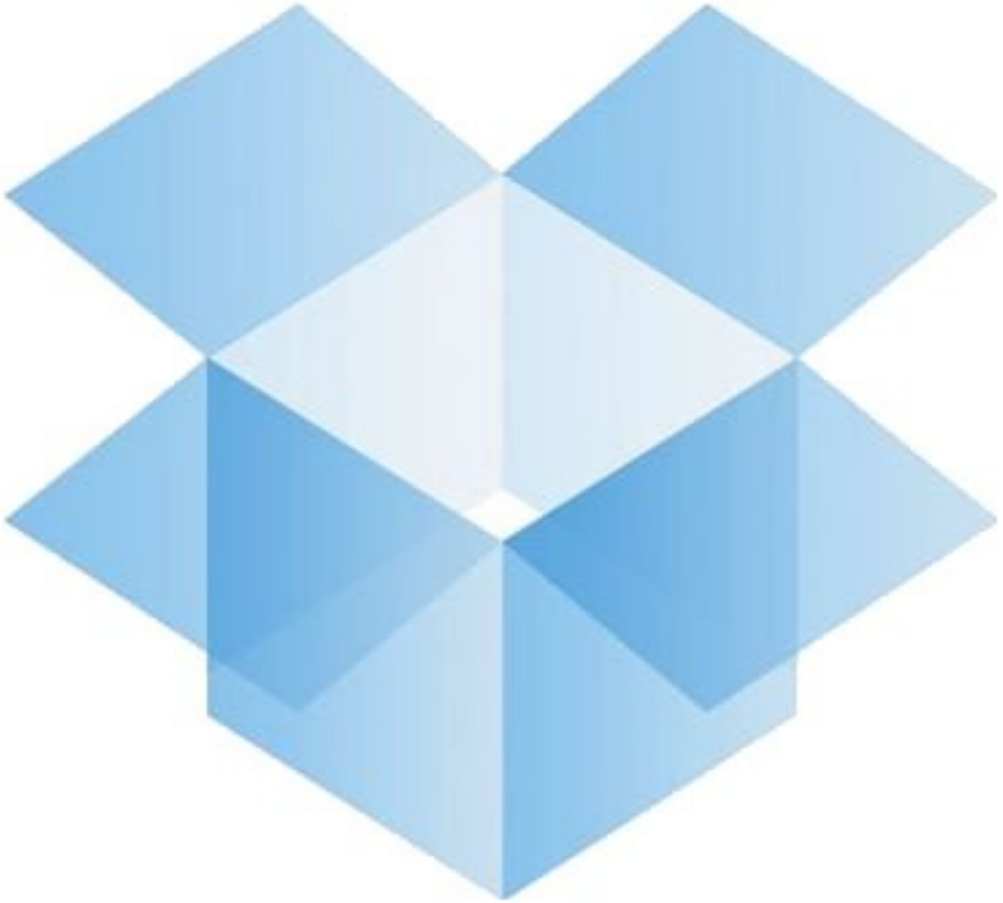
Blackberry

Palm webOS

Skype

Chrome

Dropbox



Sony Playstation

... and so forth


Route
Santa Monica Blvd
Santa Monica Fwy
10
Lincoln Blvd
Menu
1000 m
2000 m
Mio


SONY
HD
AVCHD
HANDYCAM
10.2 MEGA PIXELS
HYBRID
HDD

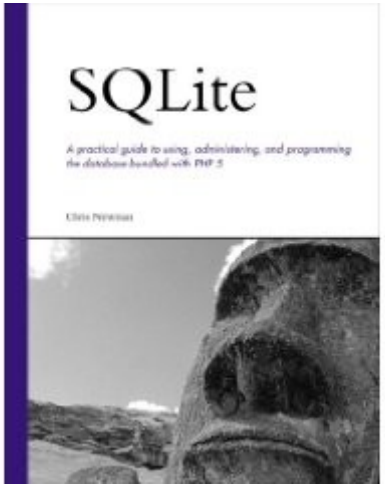
Various Programming Languages

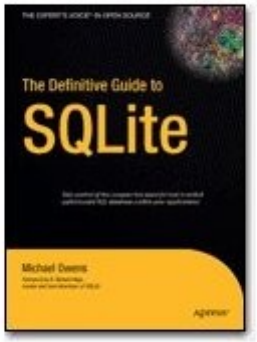
ActiveState
The Dynamic Languages Company

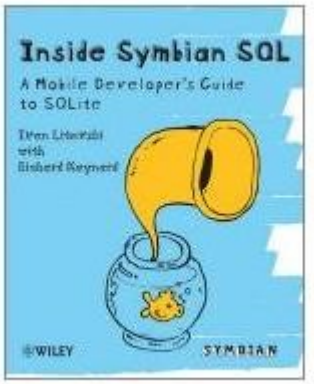
REALbasic

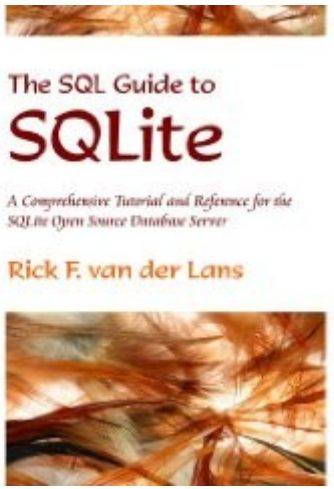
ADOBE® AIR™

Books About SQLite



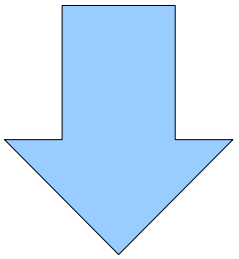






Estimating Adoption Of SQLite

- 450 million Skype users
- 300 million Firefox users
- 100 million Nokia/Symbian smartphones
- 50 million iPhones
- ??? Other active users.



500M to 1B active deployments

SQLite.org

The Company

The SQLite Development Team

Business Plan

- Annual maintenance subscriptions
- Technical support agreements
- Proprietary extensions
  - SQLite Encryption Extension
  - Compressed and Encrypted Read-Only Database
  - Test suite for embedded devices
- Training & custom development work
- Certification artifacts
- SQLite Consortium

The SQLite Consortium

- Guarantees of project continuity
- Enterprise-level technical support
- Highest priority bug fixes

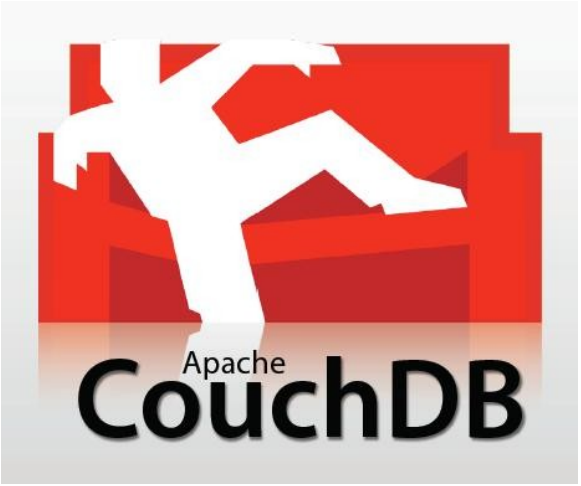
Small. Fast. Reliable.
Choose any three.

- http://www.sqlite.org/
- Per day: 16K unique IPs, 250K hits, 10GB xfer
- Hosted by Linode.com
  - Linode 720
  - Since 2004
- Custom web server on xinetd
- CPU utilization: 2.87%

? Is SQLite still relevant?

? Aren't people moving away from SQL?

? What about all these new cloud-computing databases?


Apache
CouchDB


simpleDB


mongoDB


Cassandra


Azure

The CAP Theorem

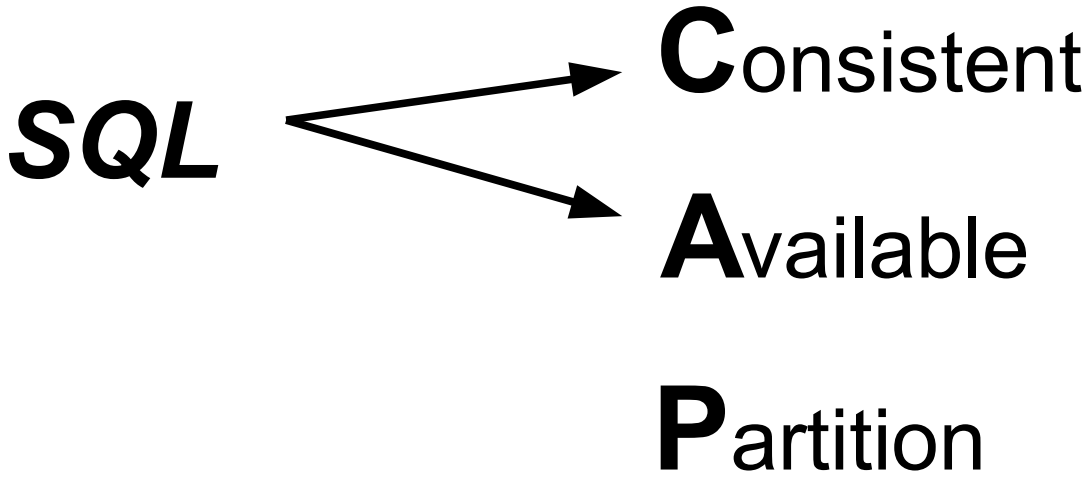
Consistent

Available

Partition

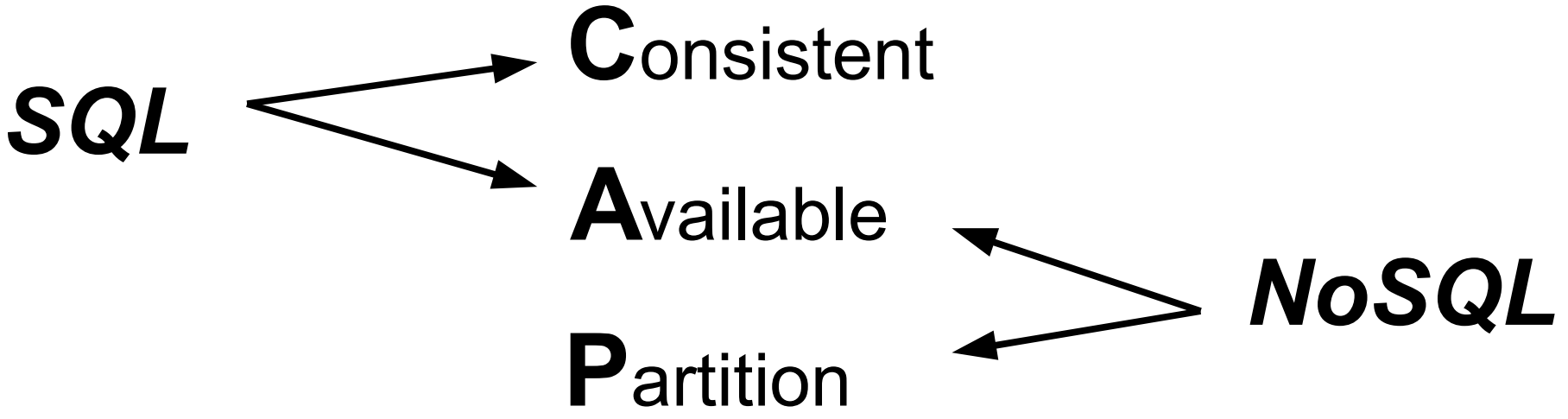
Choose Any Two

The CAP Theorem

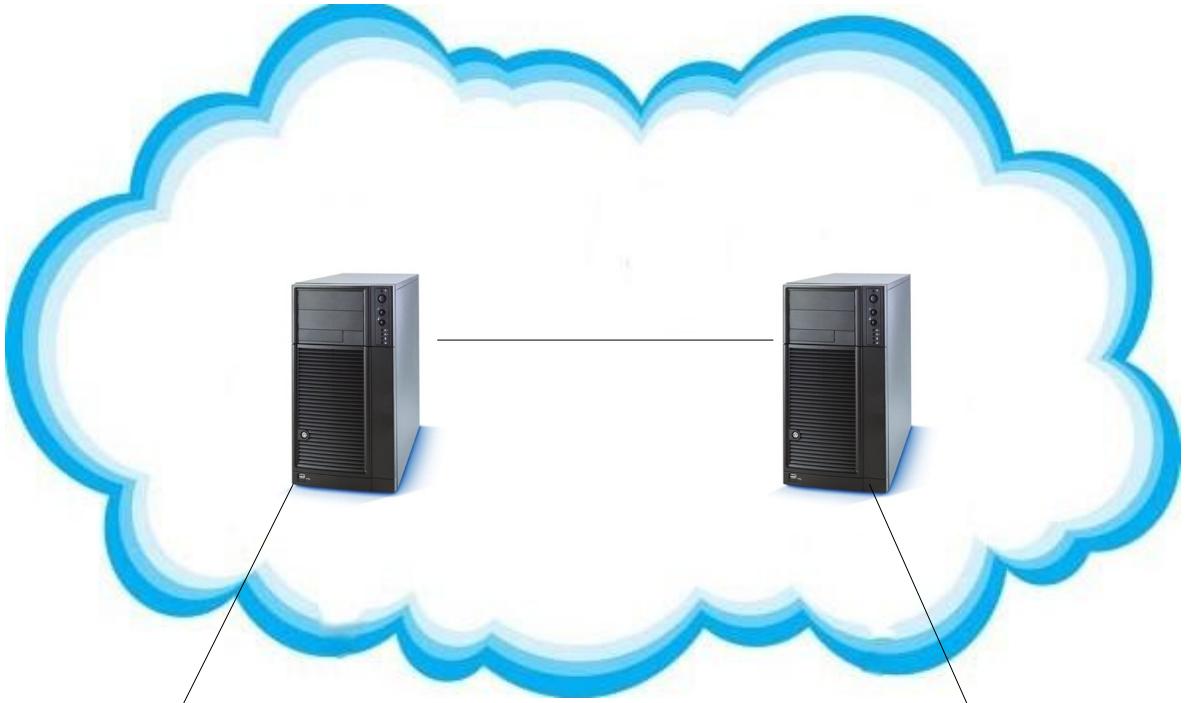


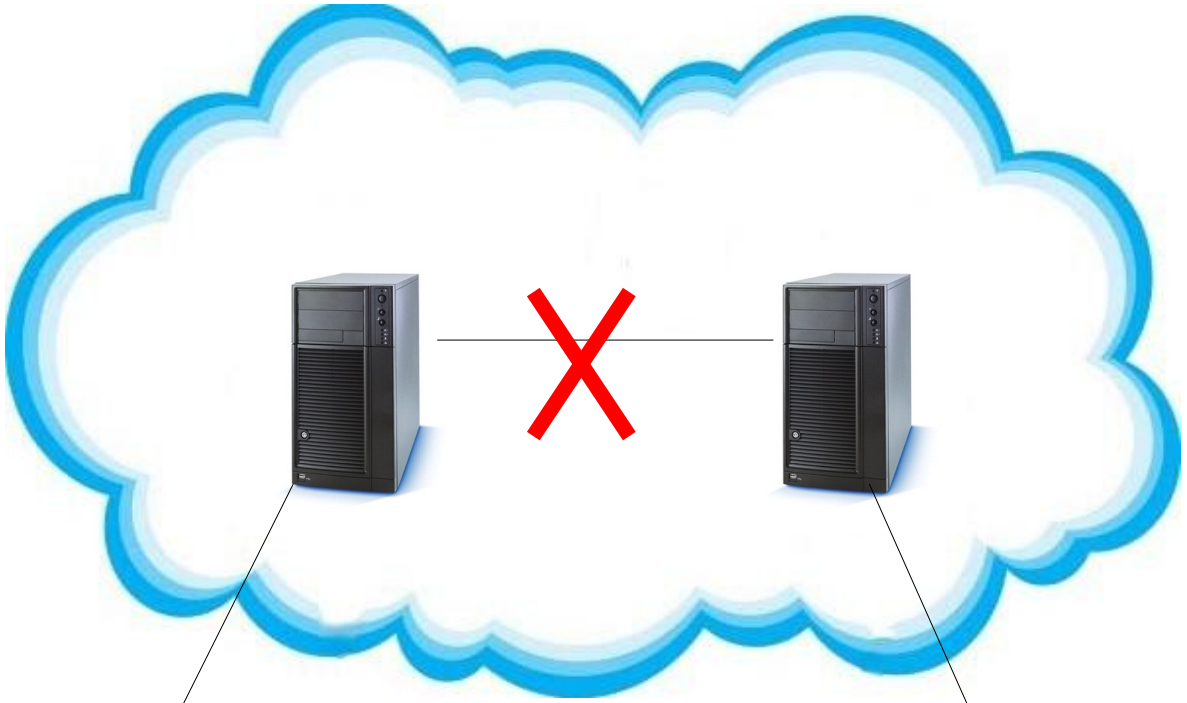
Choose Any Two

The CAP Theorem



Choose Any Two





Scaling Up versus Scaling Out

Scaling Up:

Increase the size and speed of the database server

Scaling Out:

Increase the number of database servers in your cloud

Scaling Up versus Scaling Out

Scaling Up:

SQL

Increase the size and speed of the database server

Scaling Out:

Increase the number of database servers in your cloud

NoSQL

ACID versus BASE

SQL	*NoSQL*
Atomic,	**B**asically
Consistent,	**A**vailable,
Isolated,	**S**oft-state,
Durable	**E**ventually consistent

Can We Not Keep Consistency?

- A single PostgreSQL server can handle:
  - Every seat reservation for the largest airline
  - Every book sale in the USA (online or otherwise)
- Sharding is a fallback
  - Separate database server for each airplane
  - Separate server for ranges of ISBN numbers

SQL without Consistency?

- Definitely have to give up
  - UNIQUE
  - FOREIGN KEY
- All current implementations disallow:
  - BEGIN, COMMIT, SAVEPOINT, ROLLBACK
- Most current implementations:
  - Key/Value only
  - No query language
  - No indexing, sorting, aggregating

What To Do?

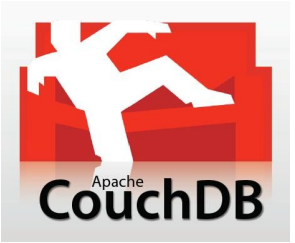
- Avoid abandoning Consistency if you do not really have to.
- Avoid regressing from high-level SQL to low-level KV databases.
- Watch the cloud-database world carefully but avoid getting sucked into the hype.

Many Processes on One Server

Multiple Datacenters

One Process

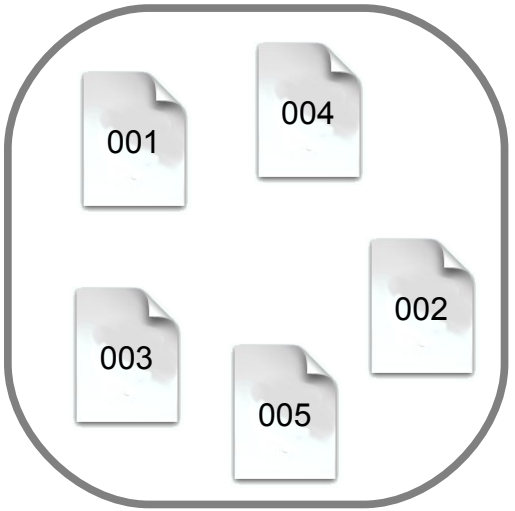
Many Servers in One Datacenter



An Example Cloud Database

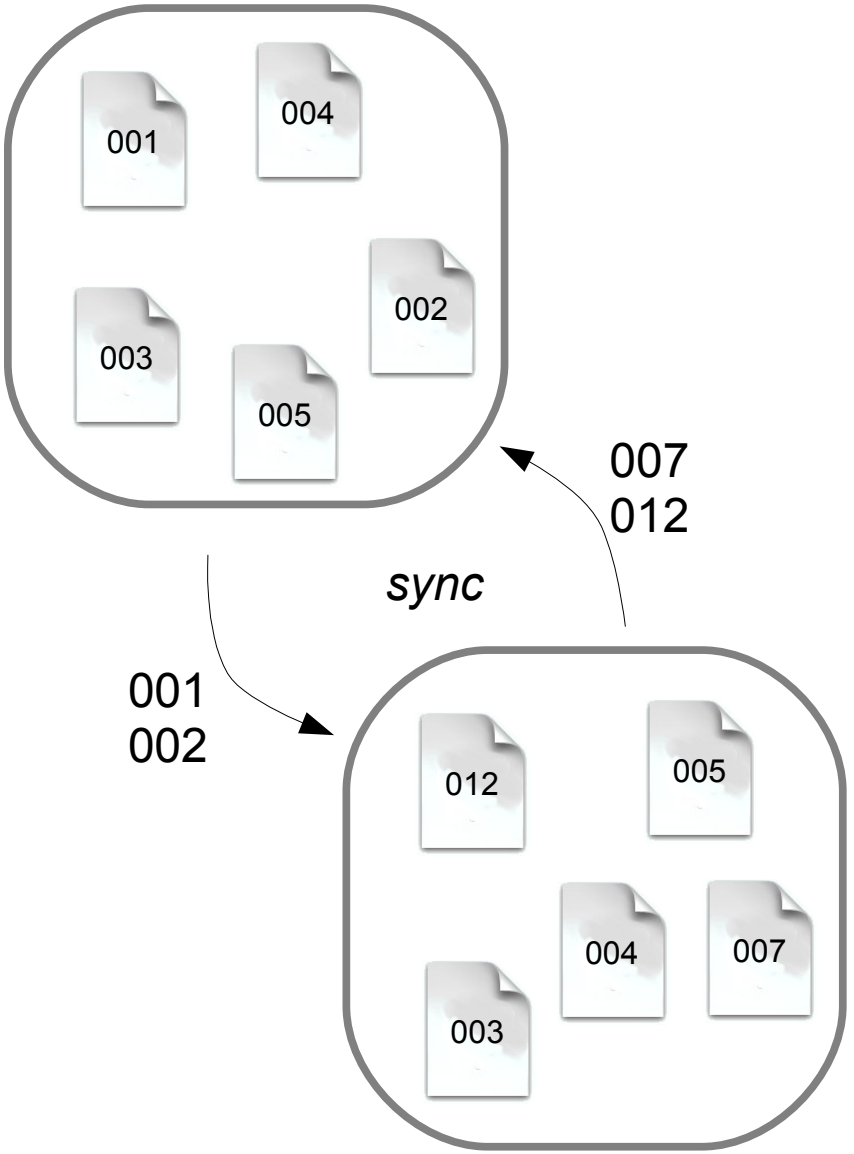
- Distributed Version Control System (DVCS)

Fundamental Concepts



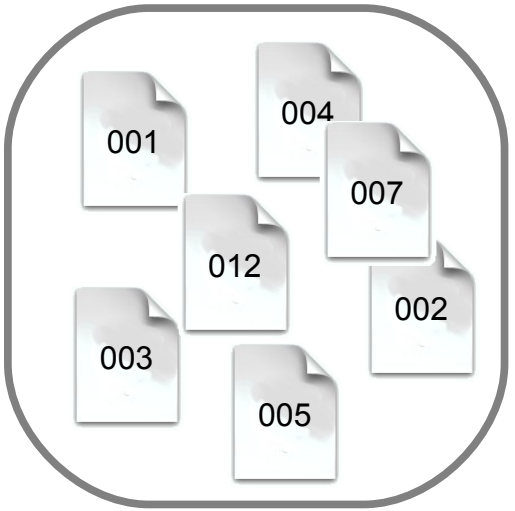
- A repository is a bag of “artifacts”
- Every users has their own repository

Fundamental Concepts

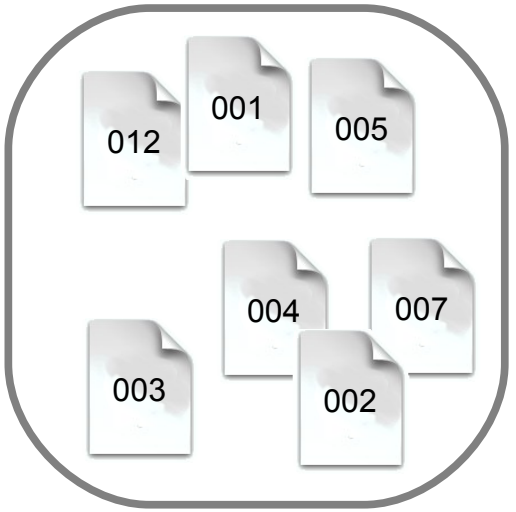


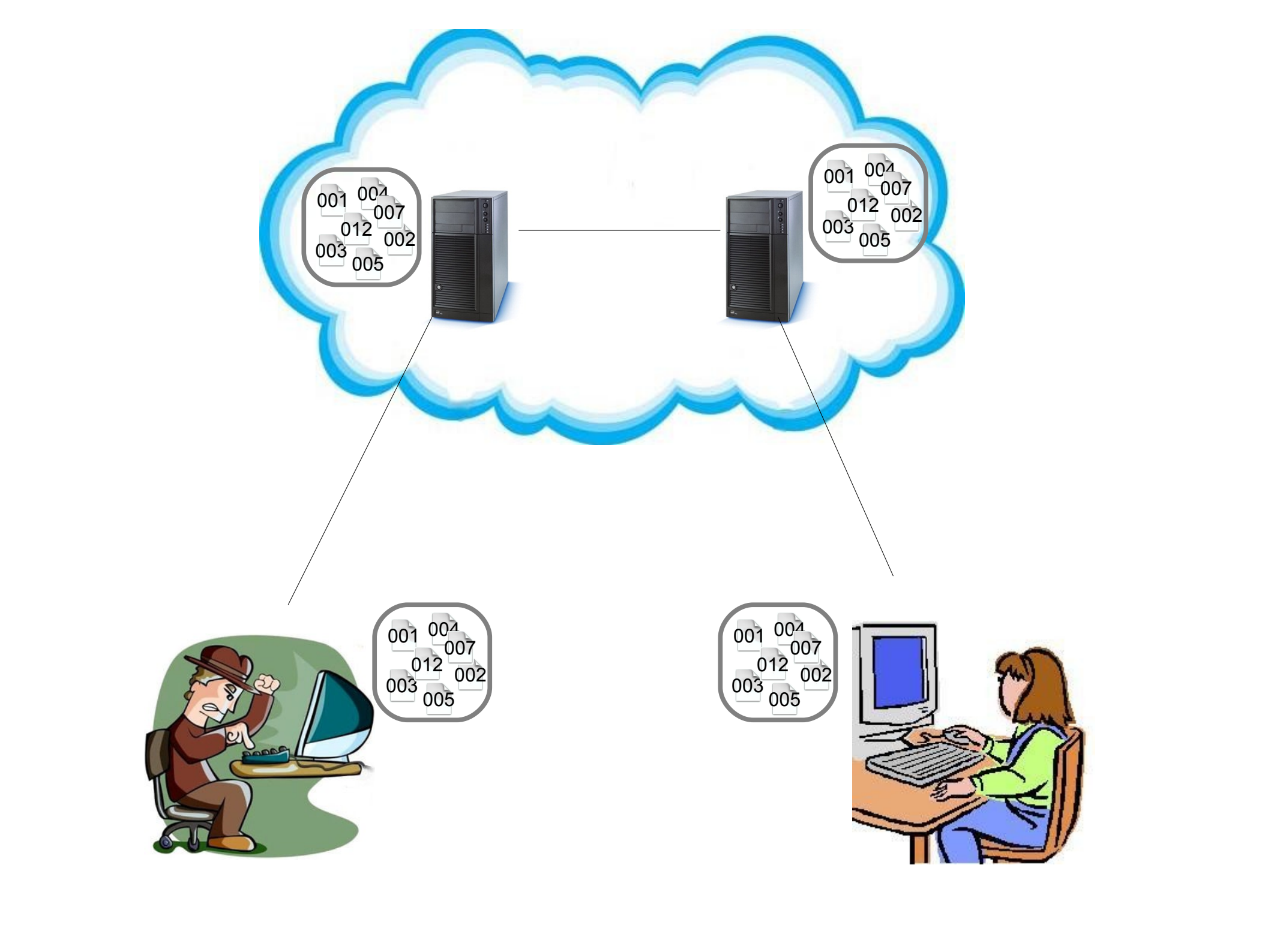
- Sync by sharing artifacts

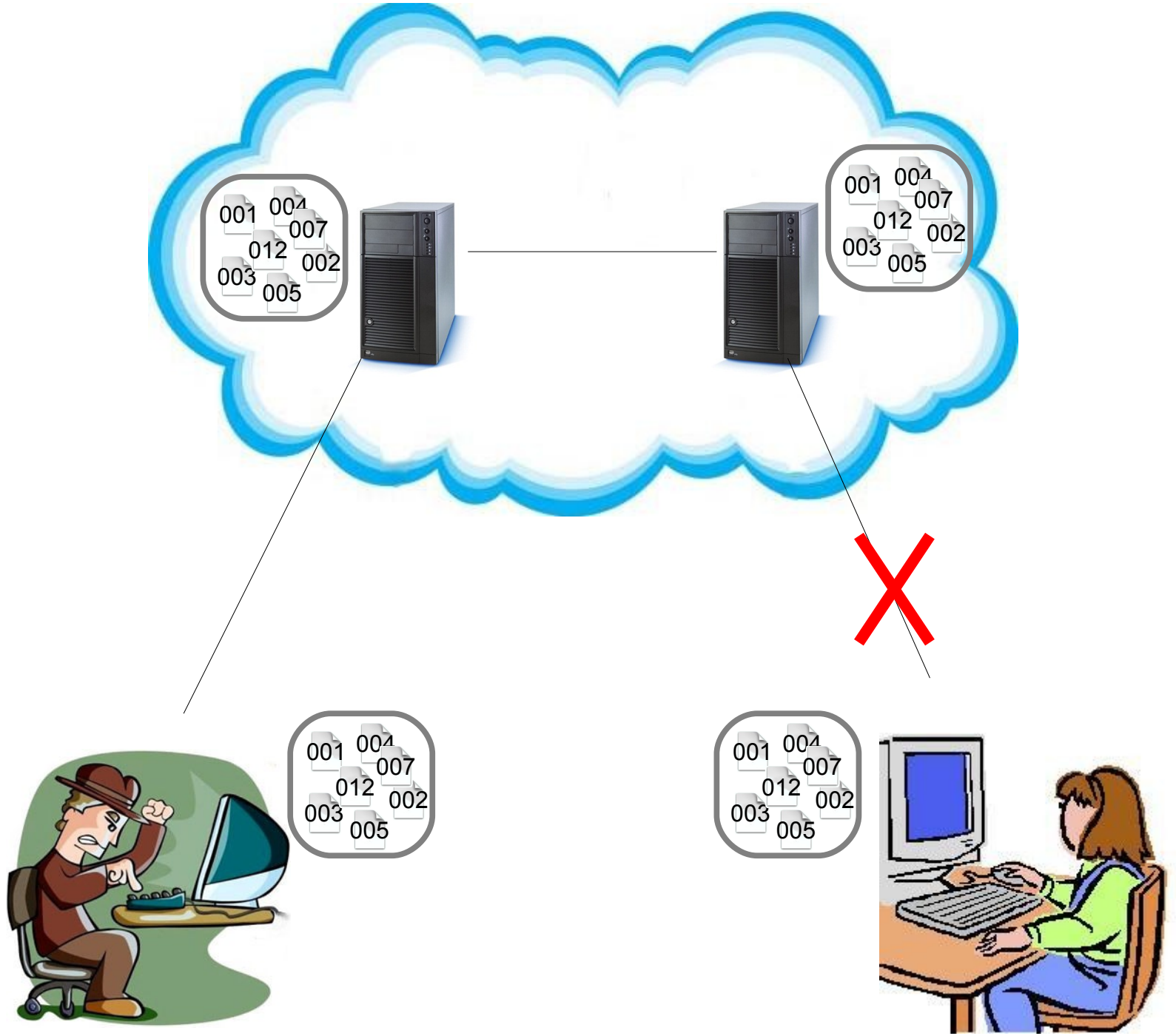
Fundamental Concepts

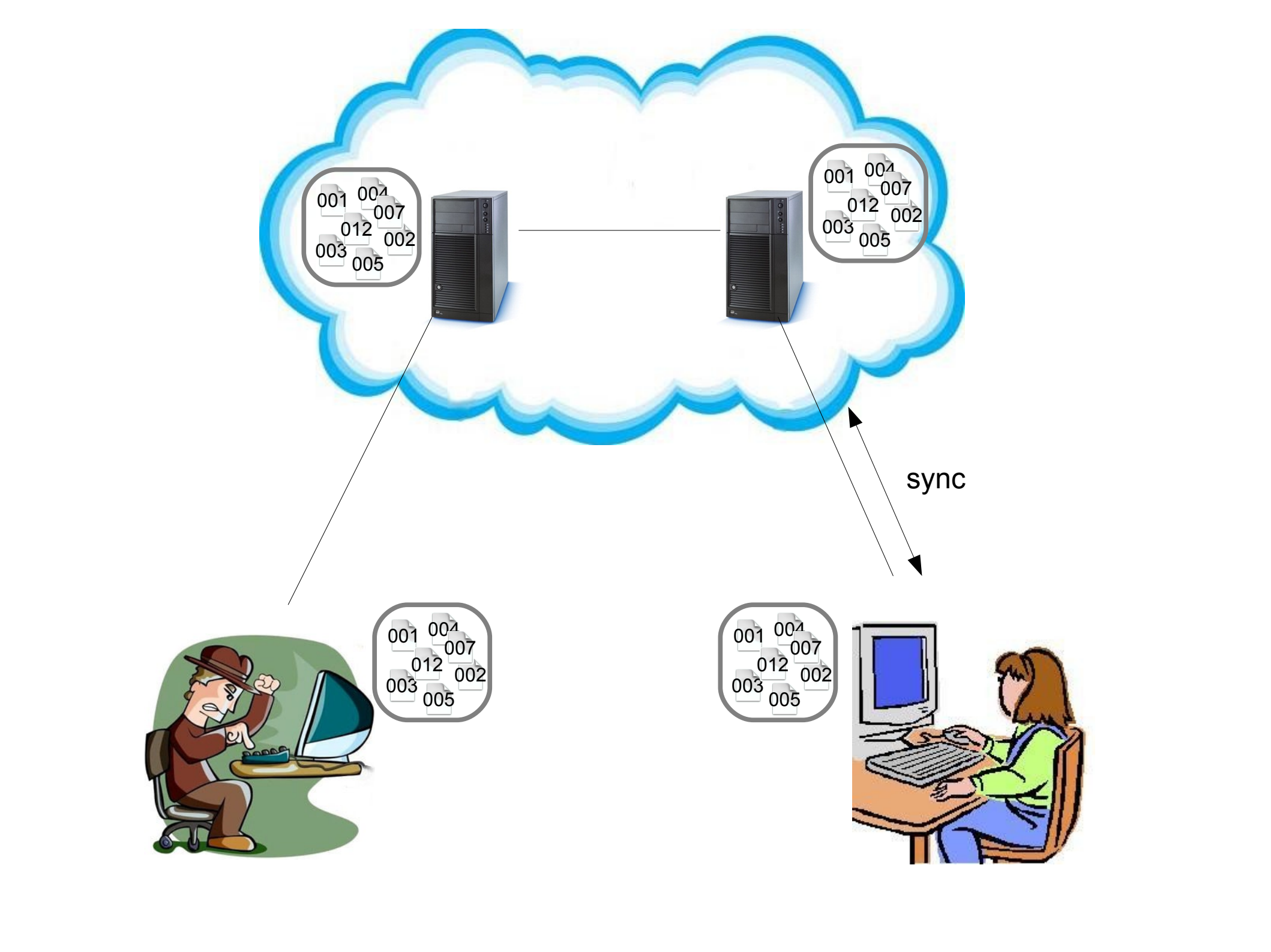


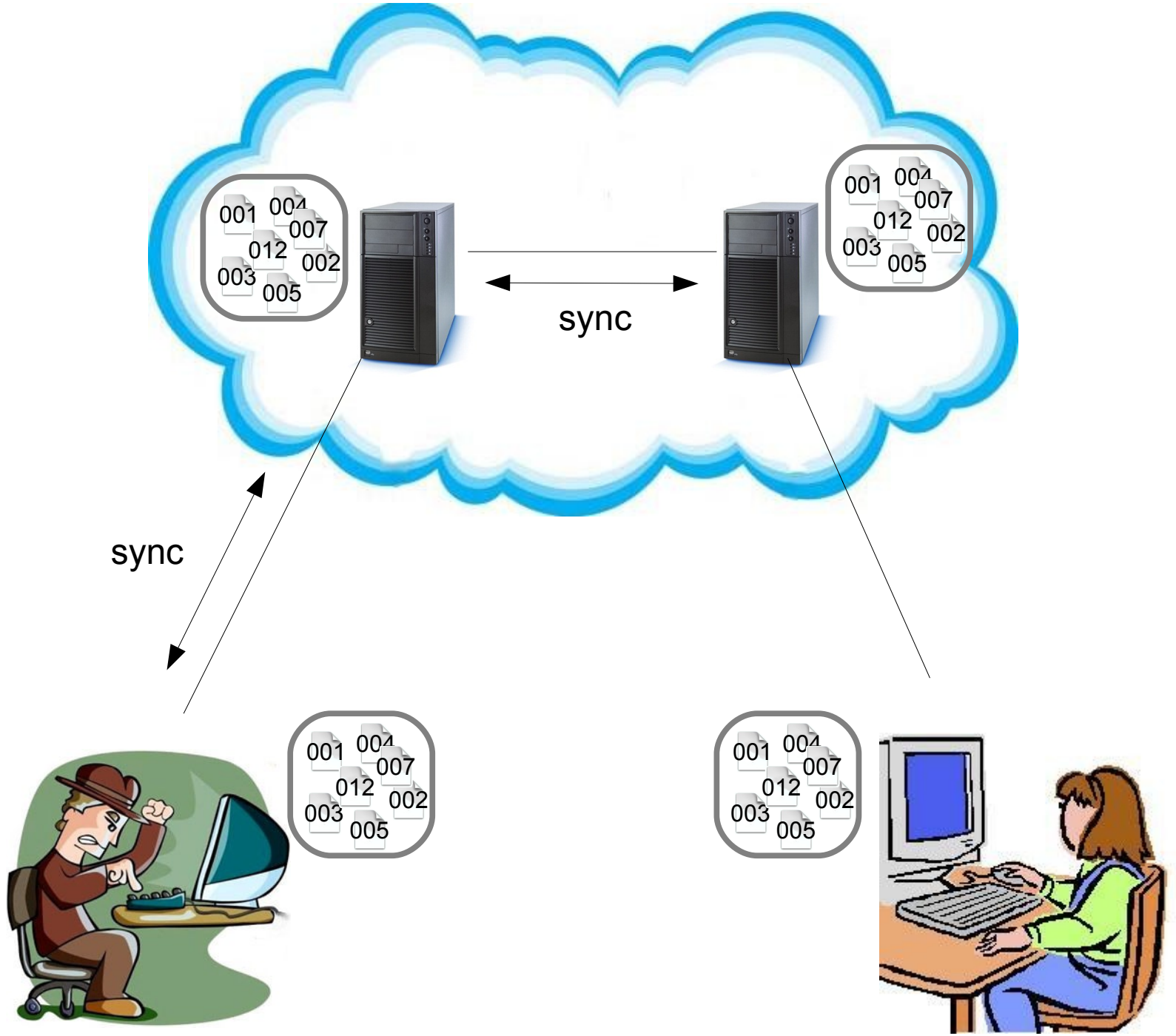
- After sync, repositories have the same set of artifacts




001
004
007
012
002
003
005
001
004
007
012
002
003
005
001
004
007
012
002
003
005
001
004
007
012
002
003
005


001
004
007
012
002
003
005
001
004
007
012
002
003
005
001
004
007
012
002
003
005
001
004
007
012
002
003
005


001
004
007
012
002
003
005
001
004
007
012
002
003
005
sync
001
004
007
012
002
003
005
001
004
007
012
002
003
005


001
004
007
012
002
003
005
sync
sync
001
004
007
012
002
003
005
001
004
007
012
002
003
005

http//www.fossil-scm.org/

- Distributed VCS
  - and wiki,
  - and bug reports
- Self-contained
- Built-in Web Interface
- HTTP & CGI
- “github in a box”

(Live Demo)

Fossil Server Setup

The actual 2-line CGI script that runs the canonical self-hosting fossil repository:

```
#!/usr/bin/fossil
repository: /fossil/fossil.fossil
```

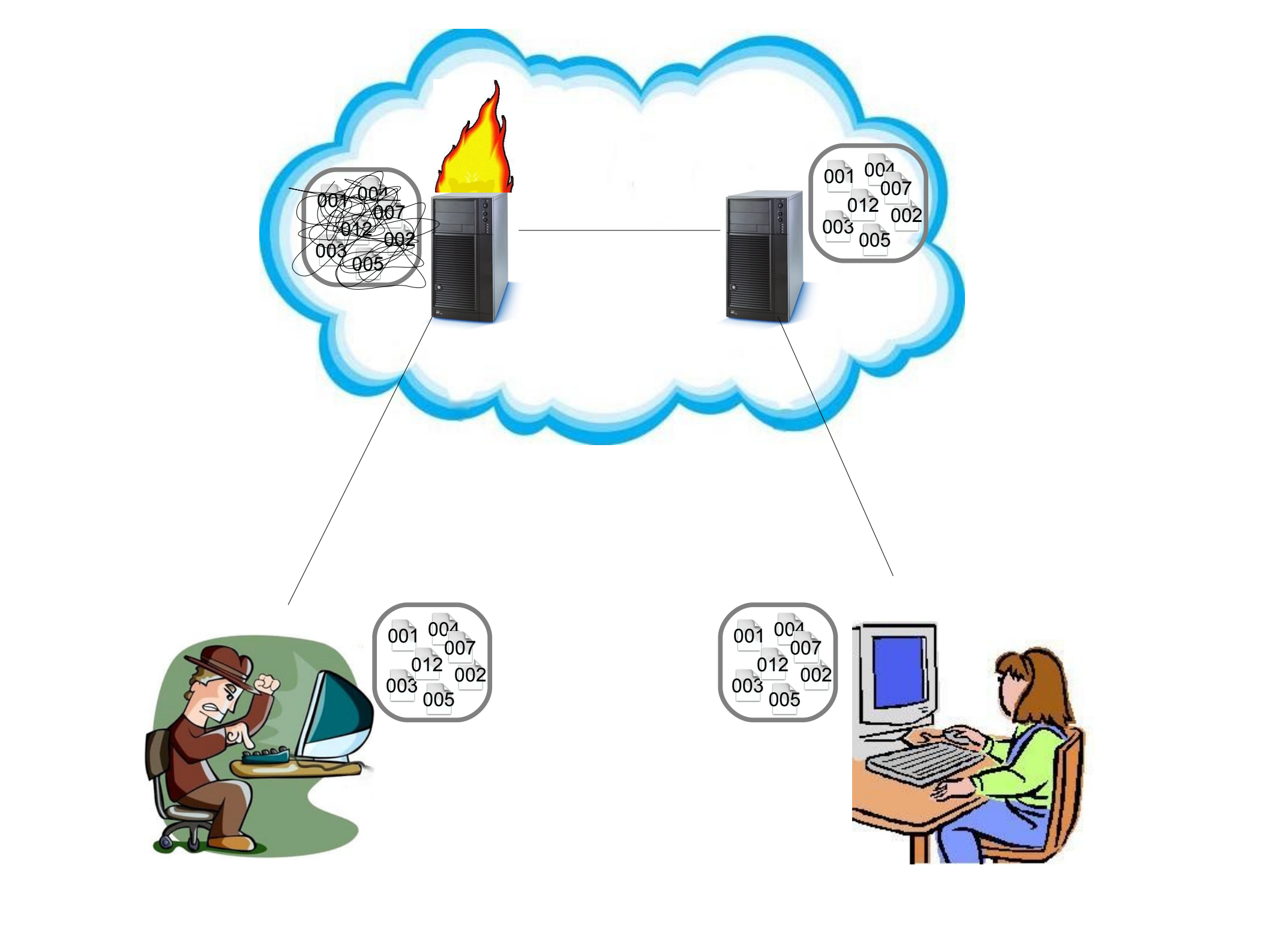
http://www.firebirdsql.org/

18-Dec-2009 Catastrophic Events at Our Internet Host

Thanks to the progressive death of the NAS server from which all our Internet facilities are served, we lost everything - Tracker, website, Wiki and news mirrors - piece-by-piece over the past days.

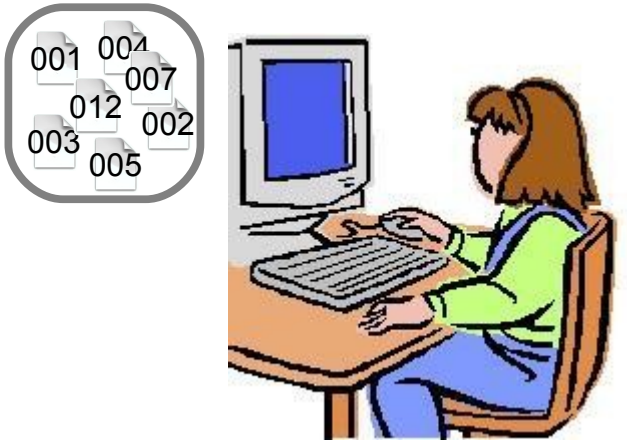
By gargantuan effort, Sean Leyne and his colleagues at Broadview have been reconstructing our infrastructure from scratch. The main website is up (obviously!). Others are more challenging, requiring radical reconstruction using new versions of underlying software, and will take time.

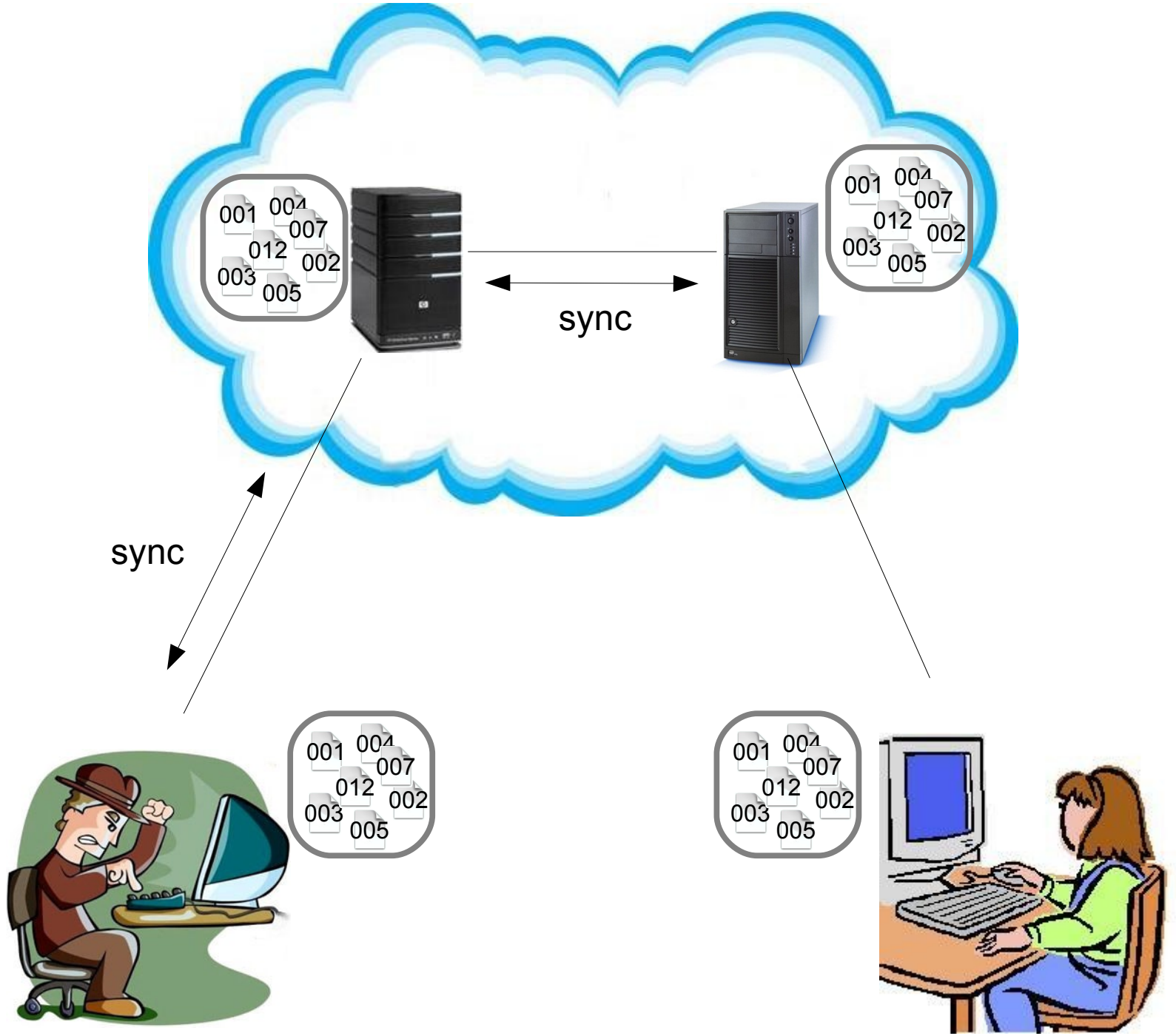
For now, if you find broken links (other than those to our other servers) then please feel free to inform us via a message to the firebird-general list (joining instructions HERE).


001
004
007
012
002
003
005
001
004
007
012
002
003
005
001
004
007
012
002
003
005


001
004
007
012
002
003
005


001
004
007
012
002
003
005


001
004
007
012
002
003
005


001
004
007
012
002
003
005
sync
sync
001
004
007
012
002
003
005
001
004
007
012
002
003
005

Fossil Is A Post-Modern Database

- Repository is not relational
- Supports Available and Partition; not Consistent

But....

- Each node stores content in an ACID SQL database (SQLite)
- Local storages includes indices for fast report generation.

Every DVCS Is A BASE Database

- For Git and Hg, local storage is a pile-of-files
  - Hand-coded read and write
  - Ad hoc format
  - What happens on a power loss?
- For Monotone and Fossil, local storage is SQLite
  - Leverage existing database engine
  - Proof against crashes
  - Simple reports

SQLite versus Pile-of-Files

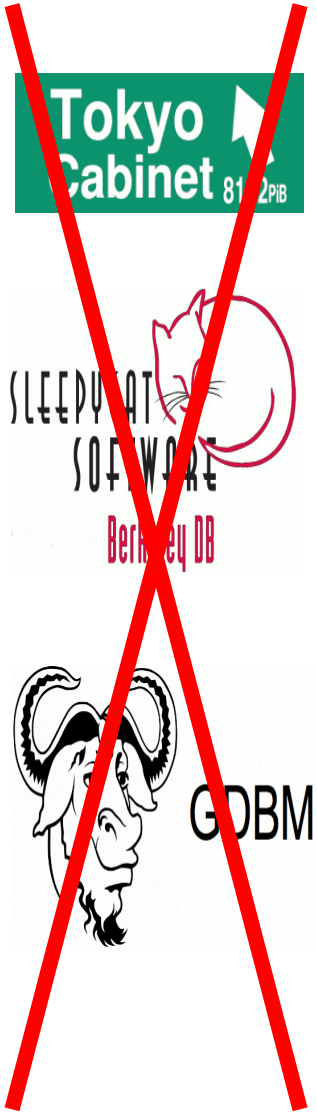
- Single-file repository
- Transactions
  - Rollback after crash
  - Rollback if self-check fails
- Viewable with 3rd party tools
- High-level query language
- Structured storage
- sqlite3_trace() for debugging


File Edit View Go Bookmarks Tools Help
http://www.sqlite.org/src/timeline?c=2000-08-17+10
Go
News Projects Talks Biz Stuff
SQLite
Source Code
Timeline
Logged in as anonymous
Home Files Leaves Timeline Branches Tags Tickets Wiki Logout
200 Events Checkins Only Tickets Only Wiki Only
10 events occurring around 2000-08-17 10:25:00.
2000-08-18
10:00:00 [e8521fc10d] Version 1.0.1 (CVS 498) (user: drh, tags: trunk)
09:58:52 [0a0576e2f9] :-) (CVS 135) (user: drh, tags: trunk)
09:34:19 [862b649204] configure script bug (CVS 134) (user: drh, tags: trunk)
09:33:40 [c773a449b1] configure script bug (CVS 133) (user: drh, tags: trunk)
2000-08-17
10:25:00 [f37dd18e3f] Version 1.0 (CVS 499) (user: drh, tags: trunk)
10:22:34 [5ec2b09478] add version numbering (CVS 132) (user: drh, tags: trunk)
09:50:00 [897b4bc0e9] allow readonly access when write permission denied (CVS 131) (user: drh, tags: trunk)
2000-08-09
17:17:25 [e8882dac23] bug fix (CVS 130) (user: drh, tags: trunk)
Done

```
INSERT OR IGNORE INTO timeline SELECT
  blob.rid,
  uuid,
  datetime(event.mtime,'localtime') AS timestamp,
  coalesce(ecomment, comment),
  coalesce(euser, user),
  (SELECT count(*) FROM plink WHERE pid=blob.rid AND isprim=1),
  (SELECT count(*) FROM plink WHERE cid=blob.rid),
  NOT EXISTS(SELECT 1 FROM plink
             WHERE pid=blob.rid
              AND coalesce((SELECT value FROM tagxref
                            WHERE tagid=8 AND rid=plink.pid), 'trunk')
                = coalesce((SELECT value FROM tagxref
                            WHERE tagid=8 AND rid=plink.cid), 'trunk')),
  bgcolor,
  event.type,
  (SELECT group_concat(substr(tagname,5), ', ') FROM tag, tagxref
    WHERE tagname GLOB 'sym-*' AND tag.tagid=tagxref.tagid
      AND tagxref.rid=blob.rid AND tagxref.tagtype>0),
  tagid,
  brief
FROM event JOIN blob
WHERE blob.rid=event.objid
 AND event.mtime>=(SELECT julianday('2000-08-17 10:25:00', 'utc'))
ORDER BY event.mtime ASC LIMIT 10;
SELECT * FROM timeline ORDER BY timestamp DESC;
```

Summary

- SQLite (and similar) for local data storage
  - Client-side
  - Consistent
  - Transactional
  - Robust
- Cling to Consistency
- Cherish high-level query languages



Embedded SQL

SQLite

Apache Derby

MySQL

HSQL database engine

Client/Server SQL

PostgreSQL

ORACLE

Informix

MySQL

Microsoft SQL Server

Post-Modern

Apache CouchDB

Cassandra

mongoDB

simpleDB

Azure

Questions & Comments