DISTRIBUTED DATABASE
FOR THE MODERN WORLD
BARREL
https://barrel-db.org
VISION AND CONCEPT
DATA IS MOBILE

"cloud" database

local database

mobile

sensor
PEER TO PEER (P2P)

discover

replicate

share
- Local first
- Put/Match the data next to you
- Query Locally
- Replicate a view of the data you need
WHAT IS BARREL
What is Barrel

- A document database
- Revision tree & atomic document updates
- Changes feed
- Replication between any nodes in both ways
- Views (~ map)
AN EFFORT TO MAINTAIN THE COUCHDB API
- view changes
- replication based on a view
- bulk get
- pluggable

WITH MODERN FEATURES
DATA are not only blobs

Replicated APPs

Couchapps but extended and revisited

REPLICATED APPS
BUILT IN ERLANG
- append only file
- MVCC
- 2 indexes (btree): by sequence, by id
- 1 index for local documents without conflict handling
- A revision tree is stored in indexes pointed to the revision offset
- The revision is stored in the file separately

**How are stored documents**
REVISION TREE

Body of leaf node in winning rev tree must be kept

Body of leaf node in losing rev trees must be kept
- Write is slow
- Read should not being blocked by writes
- No shared memory
- No atomic integer trick
- Only actors and message passing
- Operations on a doc are atomic
READ/WRITE OPERATIONS
- LRU to cache blocks
  https://github.com/barrel-db/erlang-lru
- 1 File process, Operations are limited
- DB users are linked to the database process
- Write buffer to reduce the latency
- Optional wal (barrel 014)
- NIF for the collation

**READ/WRITE OPERATIONS**
INDEX OPERATIONS

- change reader
- send /collect changes
- index editor
- get changes
- update
- share state
- View Group
- READER
- READER

DB
Credit Flow Based

The View group keep the state

View group is created on demand

kept open until it has readers

Indexer ask for updates

Read functions (Map functions) are processed in //

INDEX OPERATIONS
Create a new file to remove the fragmentation
A race between copy and the addition of new data
Require at least twice of the storage

THE COMPACTATION ISSUE
Added 2 features:

- MOVE: move doc(s) to another node or database
- User hooks functions (run in background) using hooks: https://github.com/barrel-db/hooks

- Partition on demand
- Decision depends on the application needs
CHANGES HANDLER

- Change dispatcher
- Broadcast changes
- DB
- Subscriber
changes load balancing

consumer subscribe on patterns (delete, update, …)

Create changes Load Balancer on demand

Allows remote nodes to subscribe to a queue

Based on primer (released on February 2016)
- fetch the revisions and their attachments not present on the node
- continuous or not
- try to collect multiple docs at once
- use hackney: 
  http://github.com/benoitc/hackney
- Use a Flow-based pattern instead of a classic pool

REPLICATION
Barrel
HTTPS://BARREL-DB.ORG

Enki Multimedia
HTTP://ENKIM.EU