Loose-Schema Databases and Heterogenous Data

- Doug Reeder
- Hominid Software (μ-ISV)
- Task-list management for HP webOS

Heterogenous Data

- Different communities conceptualize similar things differently
  - Music recording: artist vs. composer+performer+conductor
  - Users expect computers to adapt to them
- Single-source data can be heterogenous
  - Medical tests: any given patient has only had a few of many tests

Heterogenous Data, Homogenous Schema

- converting to homogenous format → loss or alteration of data.
  - Contact: title+given+middle+family+suffix -> given+family
  - Acceptable for 1-way transfer
- round-trip data to external sources
  - alteration generates a spurious change notification
  - losing data is unacceptable

Heterogenous Data, Manual Union Schema

- Union of all fields in any source schema
- schema must be updated before data with new source schema can be stored
- storage may be inefficient
- works fine for music meta-data, works poorly for medical tests

Loose-Schema DB

- Need not specify fields & types in advance
- Array & object fields allowed
  - If value is atomic for the purposes of the database, it doesn't violate 1NF
- Still normalize and selectively denormalize
- May still enforce field constraints
- Can't enforce foreign-key relations (no JOINs)

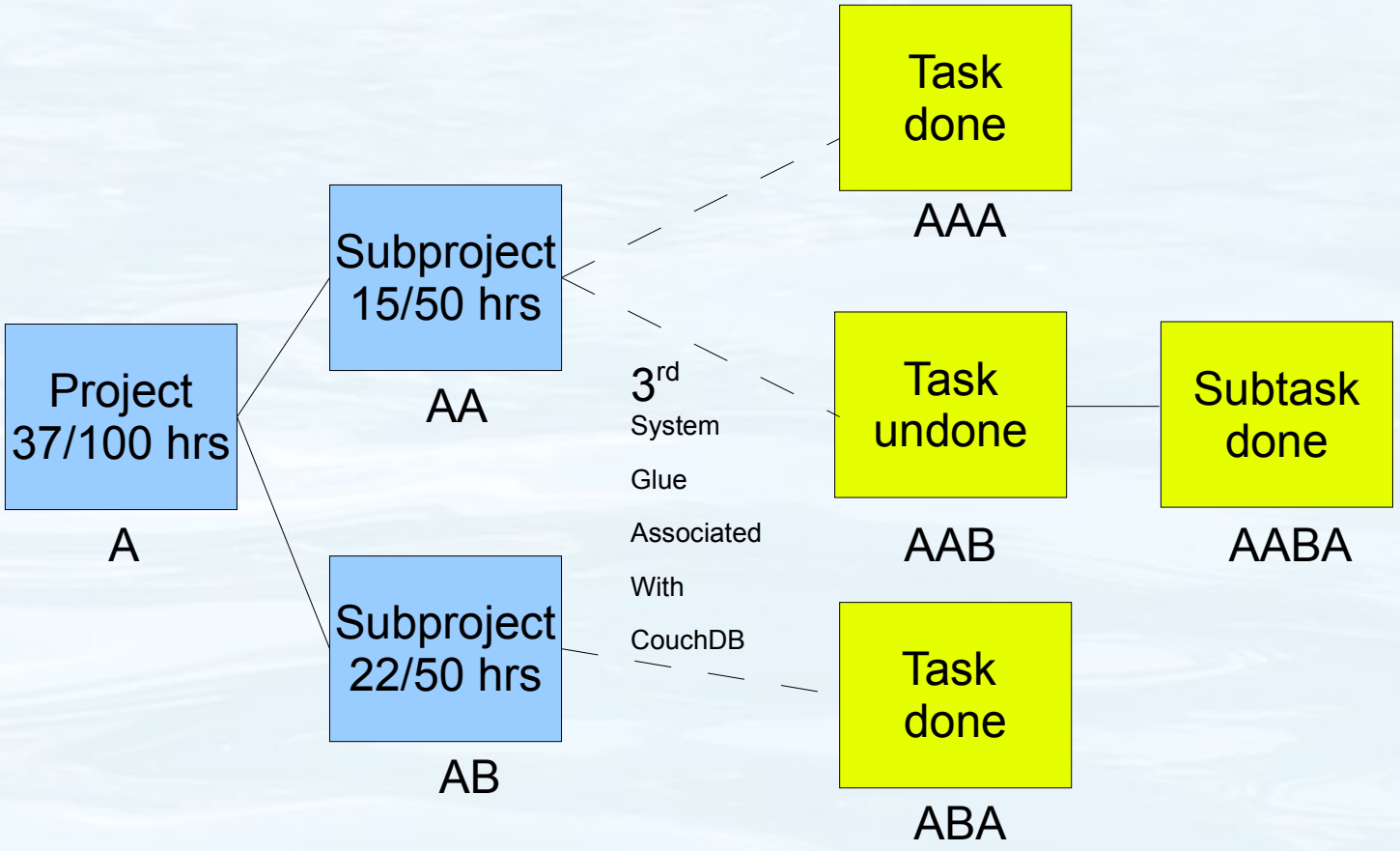
CouchDB

- stores JSON "documents"
- RESTful HTTP API + client libs
- crash-only file is always consistent
- lockless & optimistic
  - if revision number doesn't match, update fails.
  - Resolution similar to Subversion: client merge, re-submit

CouchDB Queries

- only Map-Reduce on pre-defined views
- 1st query against a view is slow; later only require incremental updating
- map func (JS or Erlang) maps document to N-dimensional array of buckets
- reduce func (optional) computes sums, averages, unions, etc over one or more dimensions

Data From Two Task Systems



Map-Reduce Completed Tasks

Map: for each ancestor path, write [# completed, 1]
Reduce: sum 1st and 2nd member of pair

path	comp.	hours	done
A	37	100	
AA	15	50	
AAA			TRUE
AAB			FALSE
AABA			TRUE
AB	22	50	
ABA			TRUE

A	AA	AAB	AB
[0, 1]	[1,1]	[1,1]	[1,1]
[1, 1]	[0,1]		
[0, 1]	[1,1]		
[1, 1]			
[0, 1]			
[1, 1]			
[3, 6]	[2, 3]	[1,1]	[1,1]

Conclusion

- Heterogenous data may be more naturally processed using a non-relational model
- couchdb.apache.org
- Questions?