

facebook

Challenges to Adopting Stronger Consistency at Scale

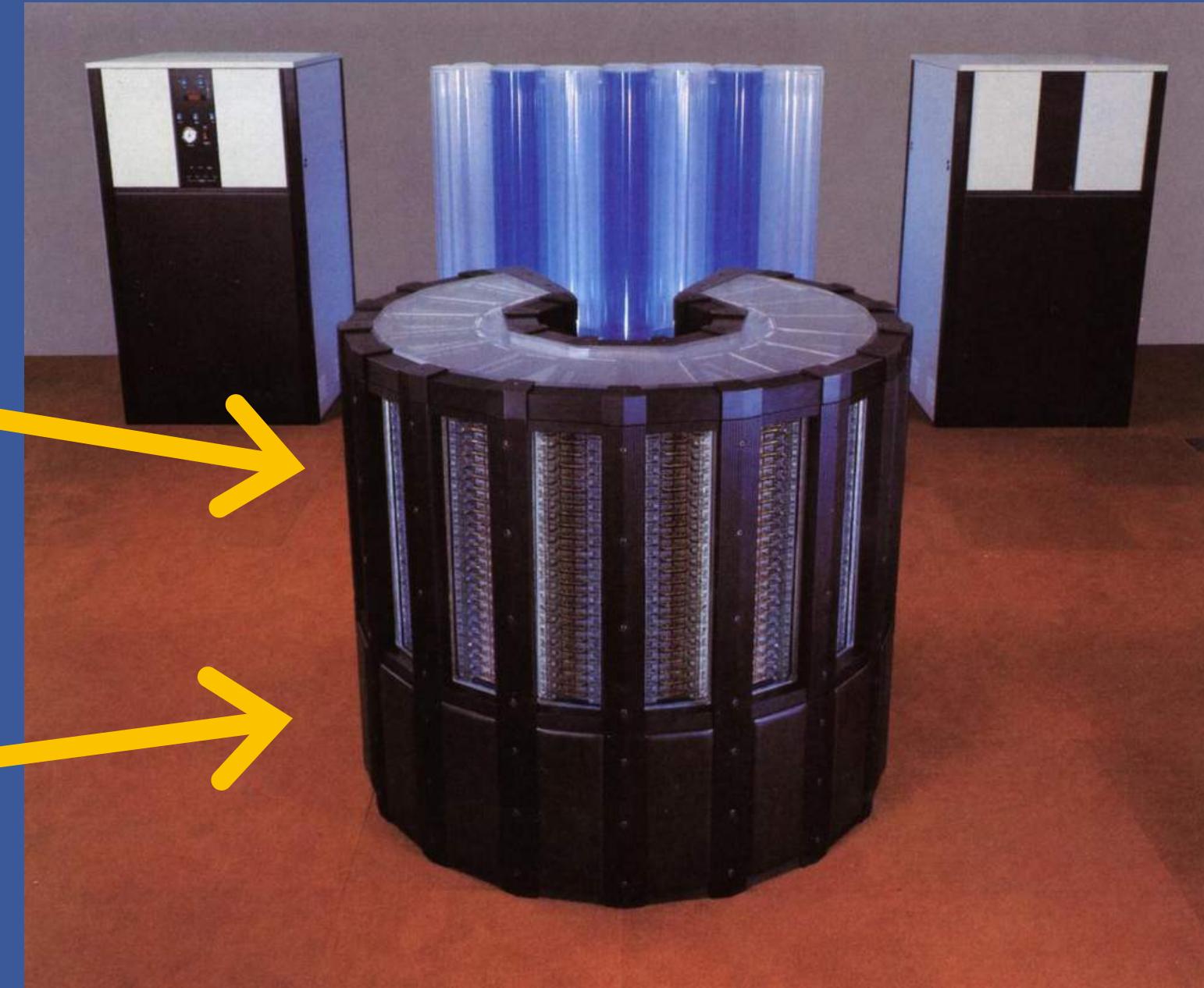
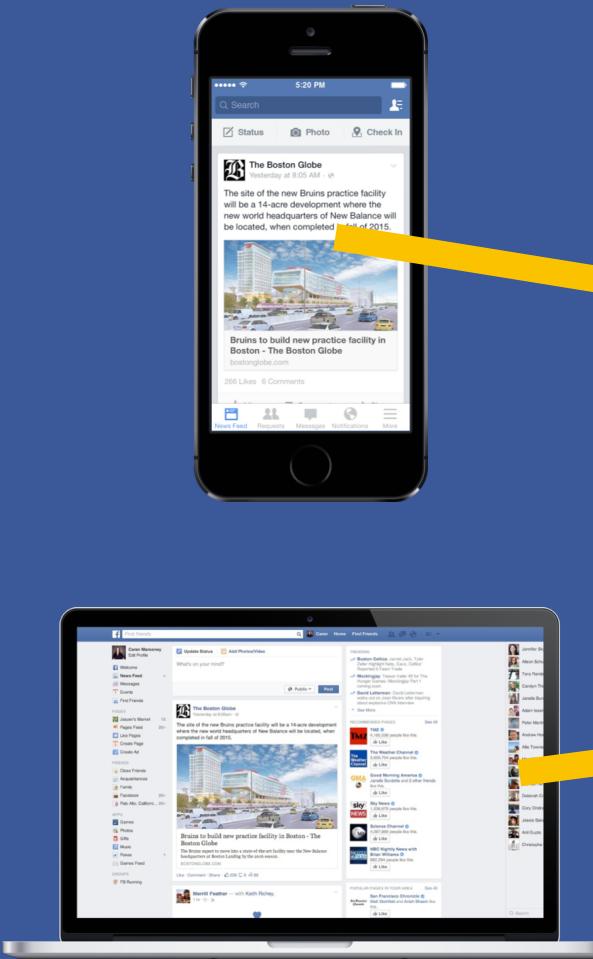
HotOS 2015
19 May

Phillipe Ajoux, Nathan Bronson, Sanjeev Kumar, Wyatt Lloyd, Kaushik Veeraraghavan

A user-visible inconsistency

The screenshot shows a social media interface. At the top, there are navigation icons: a file folder, users, messages, a globe with a red '2' indicating notifications, a search bar, and a menu. Below this is a toolbar with 'Status', 'Photo', and 'Check In' options, followed by a 'What's on your mind?' input field. A large image of a snowy road with a truck is visible on the left. On the right, a post from 'Yosemite National Park' is shown, dated '23 hrs'. The post text reads: 'The Tioga Road (continuation of Highway 120 through the park) has...'. A red arrow points from the top navigation bar to a 'Notifications' box. Inside the box, a post from 'Janet Wiener' is displayed, reading: 'posted in **Consistency Investigations.**' with a timestamp 'a few seconds ago'. A small profile picture of Janet Wiener is next to the name.

The screenshot shows a 404 error page with a red border. The main text is: 'The page you requested cannot be displayed right now. It may be temporarily unavailable, the link you clicked on may be broken or expired, or you may not have permission to view this page.' Below this is a 'Back to home' link. To the right of the error message, a large red arrow points to the right, and the text 'NOT YET ARRIVED' is written in large red capital letters.

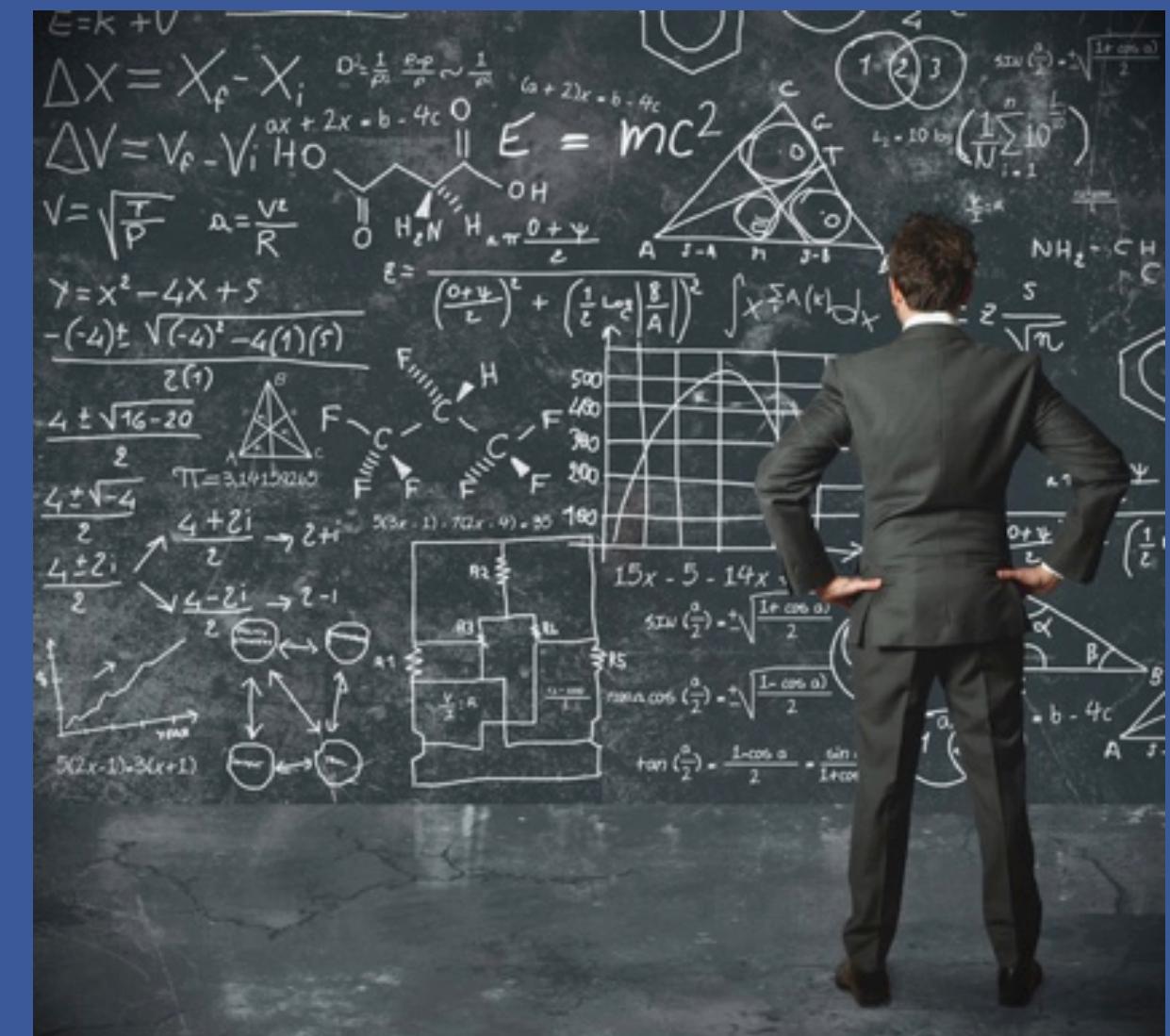


People should think FB runs on one computer

How can we fix inconsistencies?

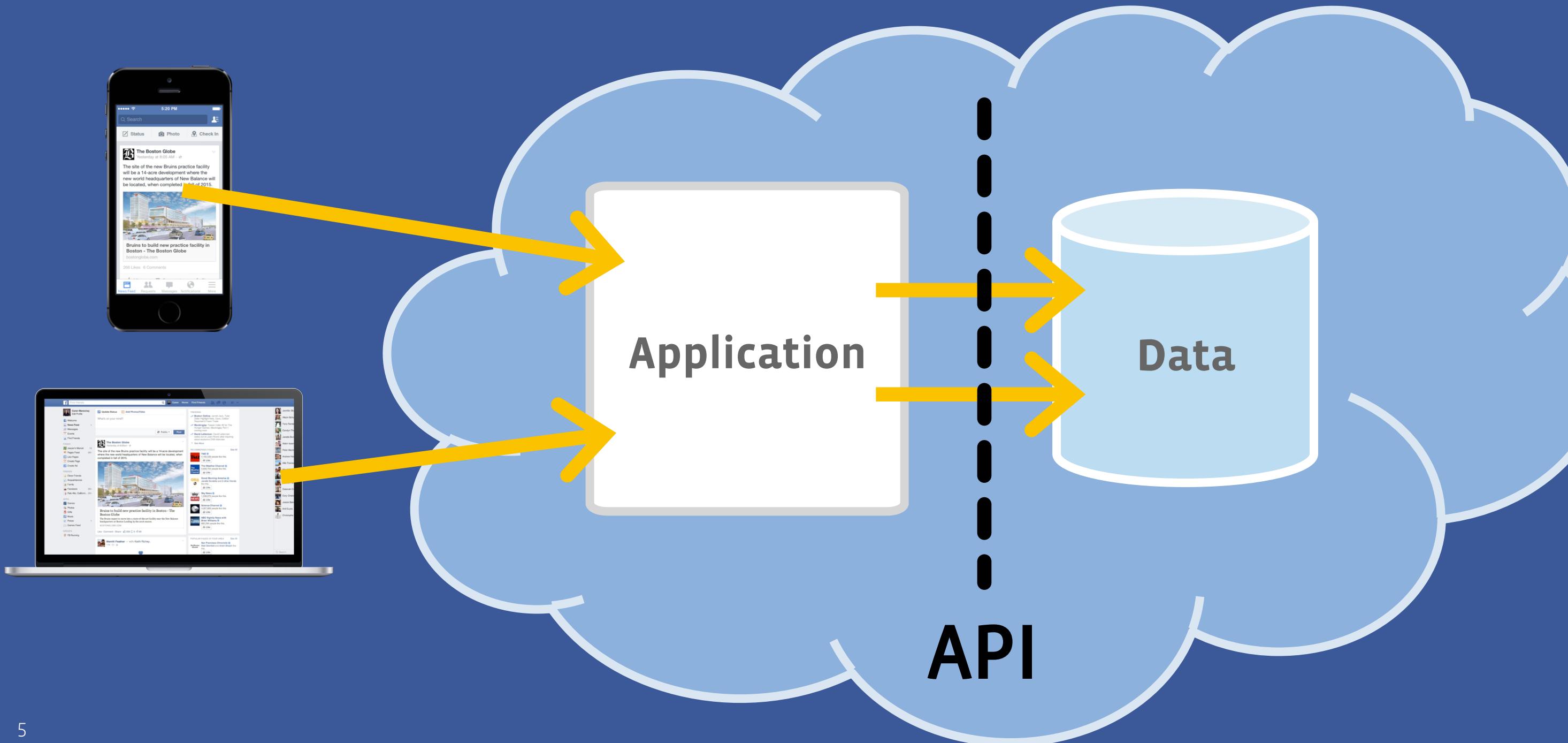


or

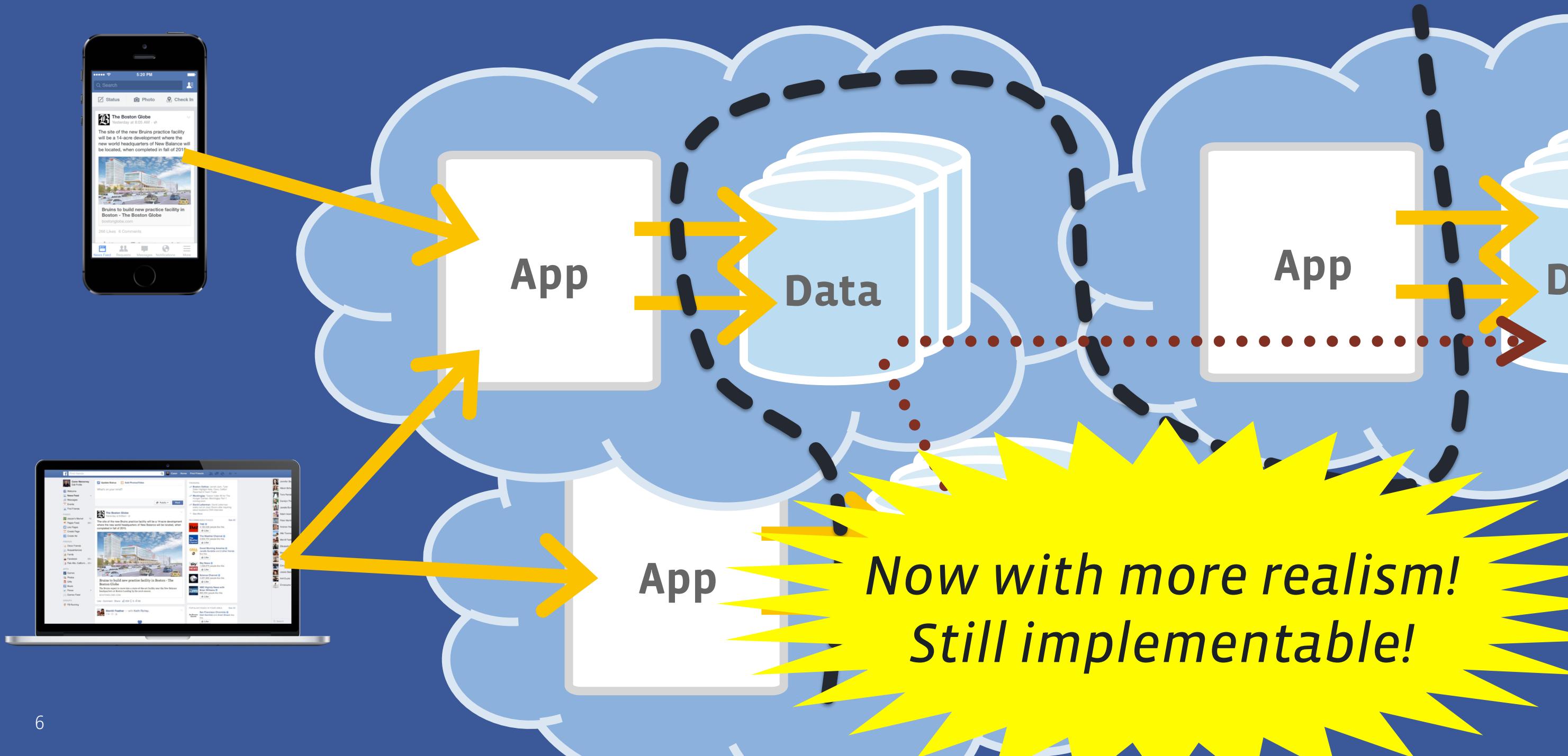


(or both)

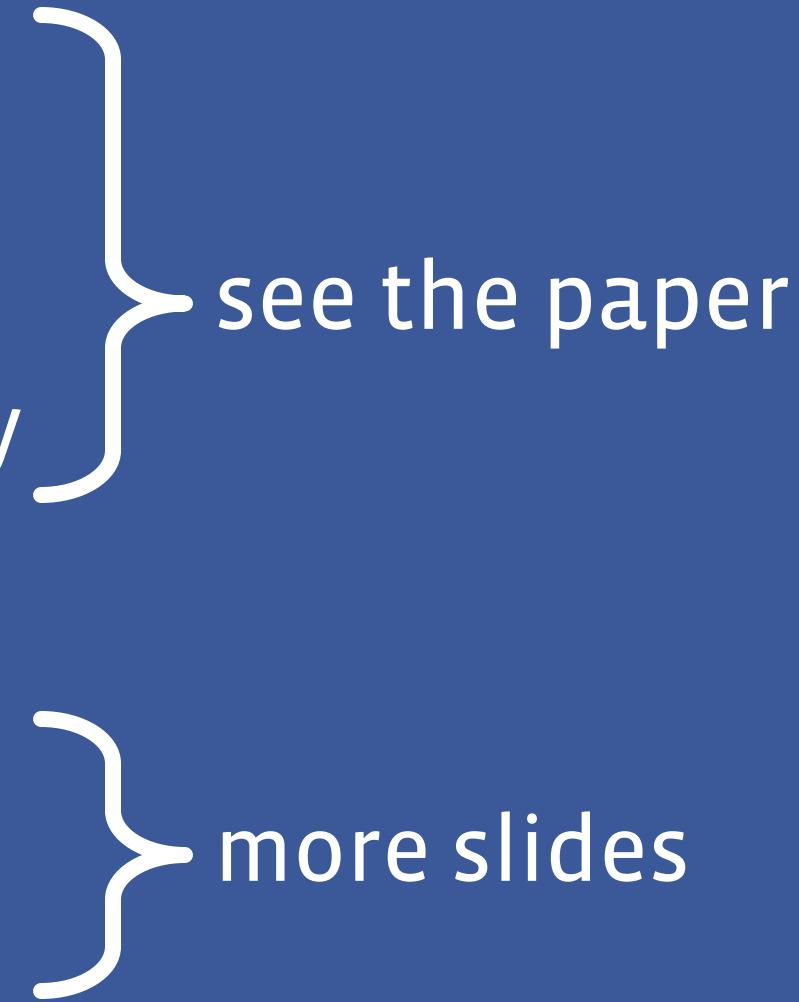
Building a site (software engineering)



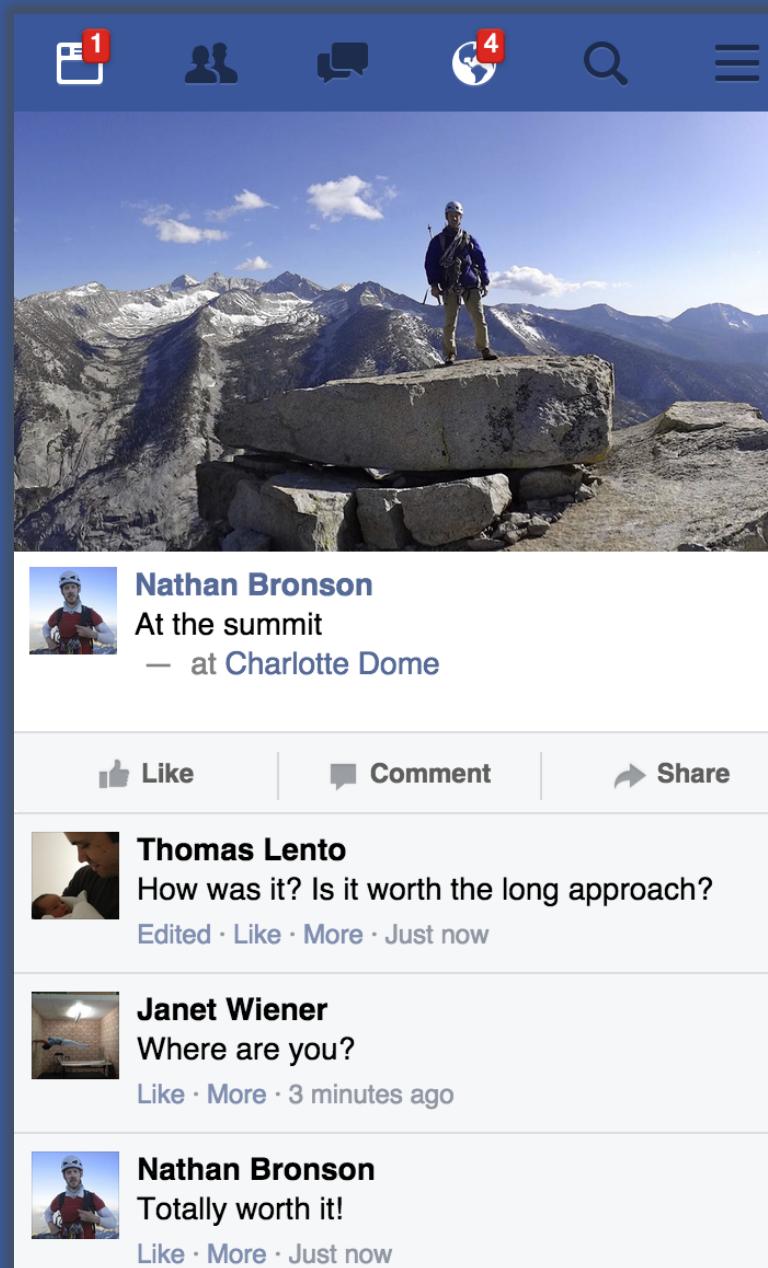
Scaling by sharding and replicating



Why not strengthen the data store?

- Will it meet our requirements?
 - Outlier sensitivity - latency & availability
 - Pathological data access patterns
 - Low average latency needed for efficiency
 - What about data copies?
 - Lots of systems store ad-hoc data copies
 - Those systems are loosely coupled
- 

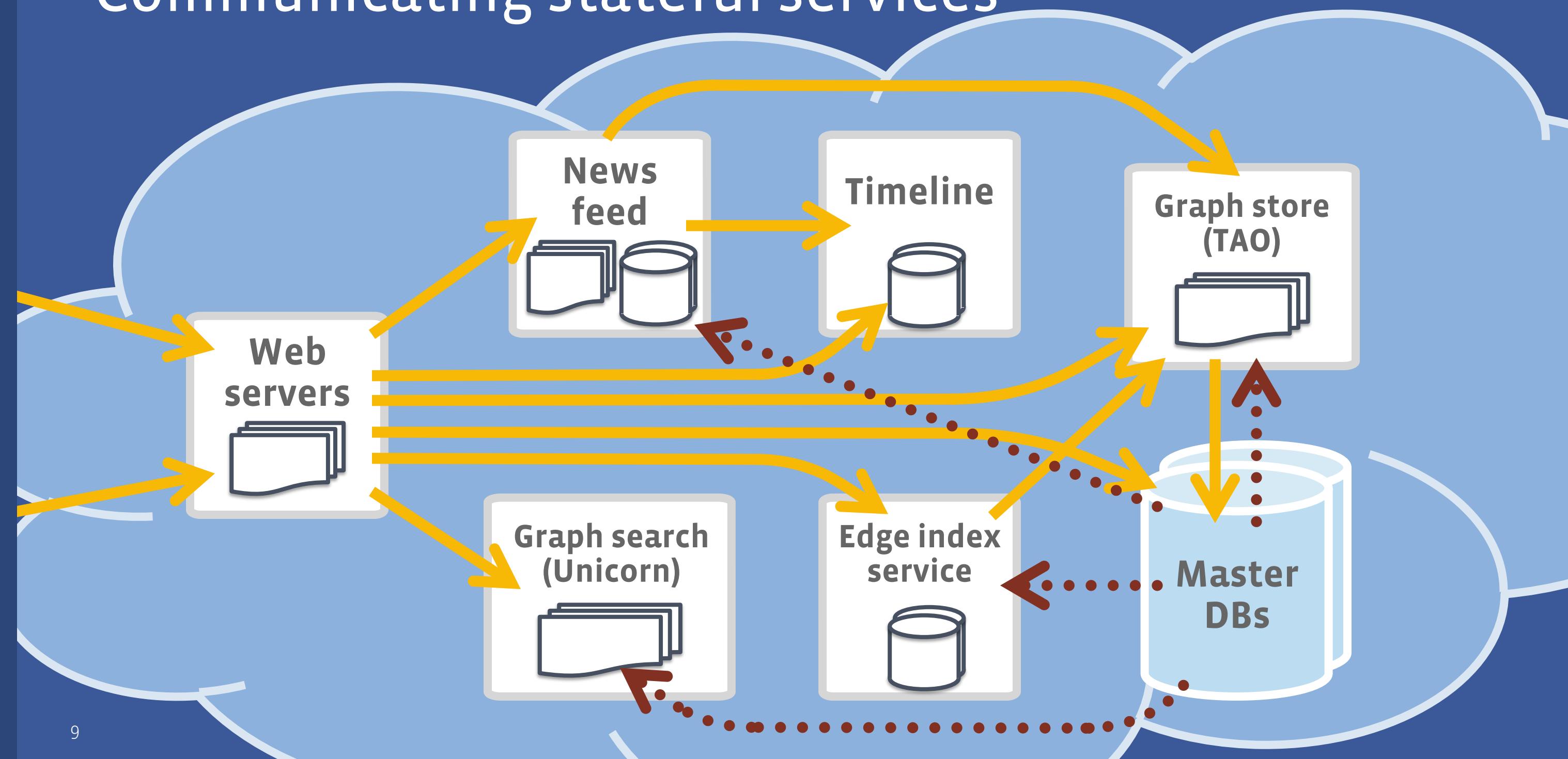
Social graph = types nodes + edges



=

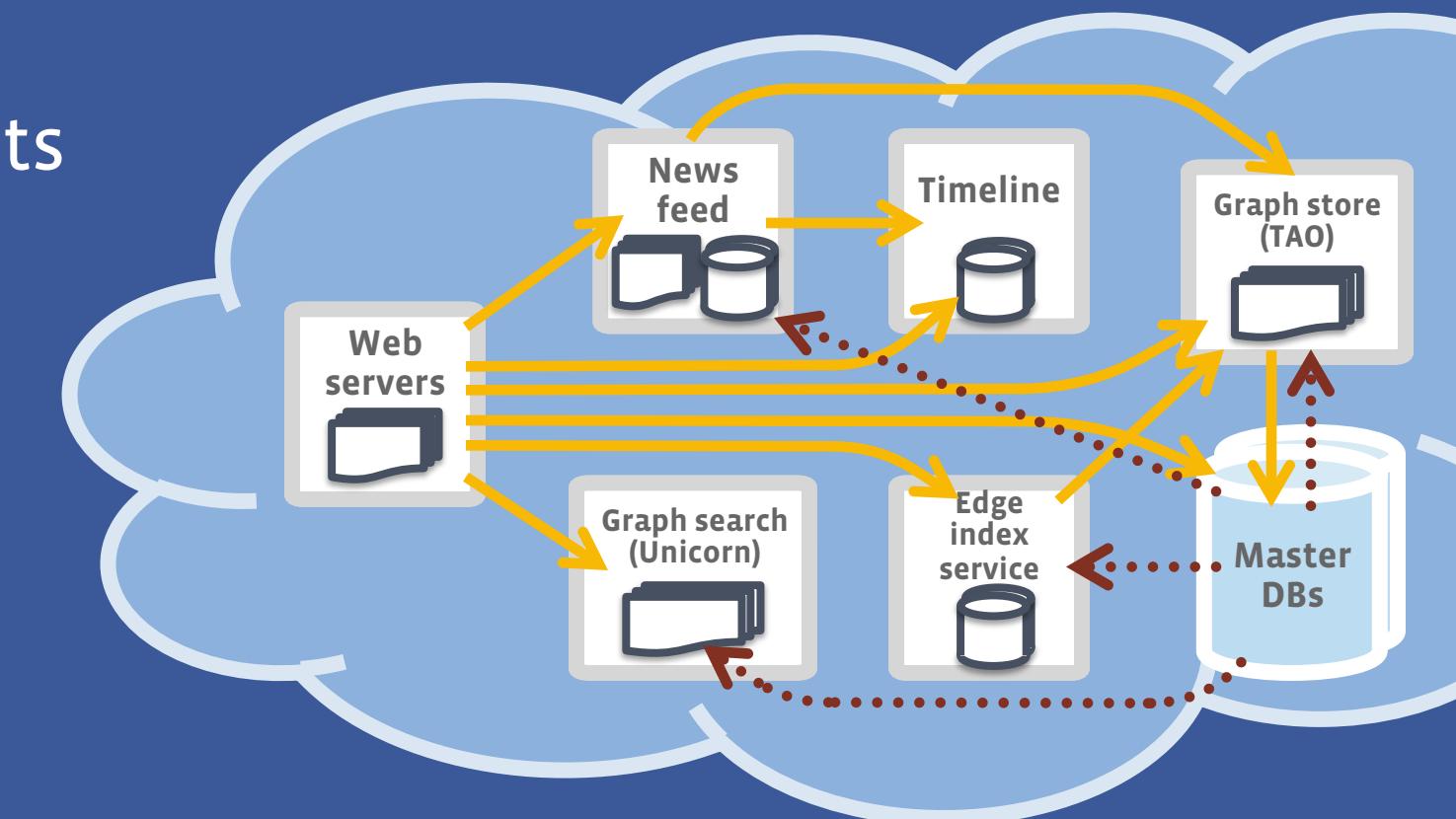


Communicating stateful services



Consistency glue challenges

- Multiple copies of the data
 - Copies are materialized query results
- Ad-hoc query languages
 - Service building block is RPC call
 - Didn't design API as a language
- Loose coupling
 - Separate teams, different languages
 - Different deployment schedules, reliability goals



Techniques from tightly-coupled databases

Locality

- ✓ Denormalization
- ✓ Caching
- ✓ Materialized join views
- ✓ Secondary indices
- ✓ Covering indices
- ✓ Partial indices
- ✓ Stored procedures

Consistency glue

- ✗ Unified analyzable query language
- ✗ Two phase locking
- ✗ Range locks, table locks
- ✗ Predicate locks
- ✗ Totally-ordered sequence numbers
- ✗ Foreign key constraints
- ✗ Linear durable log

What do I hope?

- Add tools for locality optimization to existing systems
- Make the problem topology more realistic
- Are there tools for end-to-end consistency?

facebook