Keith Adams on the Architecture of Slack, using MySql, Edge Caching, & the backend Messaging Server

Podcast with **Keith Adams** by **Wesley Reisz** on Dec 16, 2016 | Discuss

In this week's podcast, QCon chair Wesley Reisz talks to Keith Adams, chief architect at Slack. Prior he was an engineer at Facebook where he worked on the search type live backend, and is well-known for the HipHop VM [hhvm.com]. Adams presented How Slack Works at QCon SanFrancisco 2016.

Key Takeaways

- Group messaging succeeds when it feels like a place for members to gather, rather than just a tool
- Having opt-in group membership scales better than having to define a group on the fly, like a mailing list instead of individually adding people to a mail
- Choosing availability over consistency is sometimes the right choice for particular use cases
- Consistency can be recovered after the fact with custom conflict resolution tools
- Latency is important and can be solved by having proxies or edge applications closer to the user

Notes

Challenges at Slack?

Group Messaging

- 1m:30s Many companies focus on messaging; but persistent group messaging is the key focus of Slack, supporting message search and archival as well as groups
- 2m:00s Group chats in other messaging clients require you to individually add members, much like sending a group email works today

- 2m:35s Channels are used to allow optin membership of groups as well as seeing historic messages sent to that channel
- 3m:00s A slack channel feels like a place you belong in

Latency

- 3m:30s Voice and video interactions are impacted by latency; the same is true of messaging clients
- 4m:00s The user interface can provide indications of presence, through avatars indicating availability and typing indicators
- 4m:15s Latency is important; sometimes the difference is between 100ms and 200ms so the message channel monitors ping timeout between server and client
- 4m:40s 99th percentile is less than 100ms ping time
- 5m:15s If the 99th percentile is more than 100ms then it may be server based, such as needing to tune the Java GC
- 5m:25s Network conditions of the mobile clients are highly variable
- 6m:20s Mobile clients can suffer intermittent connectivity

Architecture

- 7m:15s Slack consists of a sharded LAMP stack; webservers, memcache, and a fleet of mysql instances
- 7m:30s Teams are sharded across mysql instances
- 8m:20s The realtime part of the clientserver communication is due to the messaging infrastructure
- 8m:35s Slack is a message amplifier; it takes the message written by the individual and them delivers it to all the clients that are interested in receiving the message, with the lowest latency possible
- 9m:00s The majority of desktop based connections are longlived WebSocket connections

Edge caching

- 11m:00s Users who are far away from the east coast are terminated with an edge cache called flannel (formerly slackd)
- 11m:50s The roundtrip time is much more tolerable if the edge cache serves content quicker
- 12m:15s Local conversations can be optimised with the edge cache

Posting messages

- 13m:00s Most clients use the websocket to post messages via JSON instead of using the API at api.slack.com
- 14m:00s Write amplification happens inmemory in the Java process to deliver messages to currently connected clients, and then sends the message backend
- 15m:00s There is a possibility of failure, in that the Java process may deliver the message to the network clients but then fail to persist it
- 15m:10s The platform is being redesigned and will hopefully address in future
- 16m:00s There's no evidence that this has hit people

Business and community

- 20m:00s Commercial users of Slack need to be more tightly controlled and defined, or to selectively enable/disable features for individual users
- 20m:30s Lots of users have their own logins for each service; there's interest in improving that while still allowing commercial companies to use single sign on solutions

MySql and persistence

- 21m:30s MySQL has replication and data protection built in; other companies have thousands of man years in operating without data loss
- 22m:15s Users care that persistence works and they don't lose data, not what the storage system is
- 22m:40s Lots of the data is relational but consistency is not absolute; master to master replication allows for eventual (in)consistency
- 23m:40s The best order fit for the master to master is to selectively pefer which master is written to using the loworder bit of the team identifier; so even teams prefer to write to one master and odd teams will prefer to write to the other master
- 24m:30s Availability is being preserved instead of consistency in the CAP triangle
- 24m:55s Insert on duplicate key update semantics allows users to post messages, and if the message has been replicated previously then the subsequent insert will overwrite it

Consistency and conflicts

- 25m:15s Consistency problems can occur when two rows are inserted in the two masters simultaneously; it is a querybyquery case that needs to resolve conflicts in an appropriate way

containers, and more.

Microservices: From Design to Deployment (NGINX published)
Microservices architecture is the new state of the art in application development and deployment. This ebook provides a useful set of resources to get started in your transition to microservices.

RELATED CONTENT

Modern Web Security, Lazy but Mindful Like a Fox Dec 16, 2016

Julien Nioche on StormCrawler, Open-Source Crawler Pipelines Backed by Apache Storm
Dec 15, 2016

Ashley Nolan Surveys State of JavaScript Tooling in 2016 Dec 15, 2016

Using a Skill Matrix for Growth and Learning
Dec 15, 2016

Angular 2.3.0 Released; Naming Guidelines Explained Dec 15, 2016

Spinnaker - Land of a 1000 Builds
Dec 15, 2016

- 26m:15s Manual conflict resolution indicates an application error in not being able to resolve conflicts itself
- 26m:35s Relaxing consistency helps availability for the system
- 27m:00s Most mutations that happen in Slack are performed at human scale and pace
- 27m:10s It's unlikely that a user will update the profile picture in a smaller number of microseconds to end up in an inconsistent state
- 27m:25s It's extremely rare that it happens, and if it does, the user can always set their picture again
- 28m:10s If there was no conflict resolution then the masters could diverge
- 28m:15s There is a conflict resolution system recipe; masters live for a month and then new read replicas are attached and caught up; when they are, they become the new masters since they are in sync with each other

MySql and the future

- 29m:00s MySql is used because Slack has operational experience and the fact that relational queries are used means that other solutions like Cassandra haven't been explored yet
- 30m:10s Slack's architecture is still evolving and it may change in the future
- 31m:30s As the growth continues and the orders of magnitude increase, there may be rewrites in the future as well

Origins of Slack

- 32m:20s Slack started as a company called TinySpec which created a massively multiplayer game called Glitch, and weren't getting the growth in the game that they were looking for
- 33m:00s The game server had a bot which indexed all messages that had been sent
- 33m:30s Users were using the builtin IRC server for messages
- 33m:50s The developers pivoted and came up with the idea of using the IRC server as a standalone product; SLACK, with a backronym of Searchable Linked Archive of Company Knowledge
- 34m:50s Group messaging succeeds if the users feel like they are part of a shared space

Companies mentioned

- Slack
- Facebook

About QCon

QCon is a practitioner-driven conference designed for technical team leads, architects, and project managers who influence software innovation in their teams. QCon takes place 7 times per year in London, New York, San Francisco, Sao Paolo, Beijing, Shanghai & Tokyo. QCon San Francisco is at its 10th Edition and will take place Nov 7-11, 2016. 100+ expert practitioner speakers, 1300+ attendees and 18 tracks will cover topics driving the evolution of software development today. Visit qconsf.com to get more details.

More about our podcasts

You can keep up-to-date with the podcasts via our RSS feed, and they are available via SoundCloud and iTunes. From this page you also have access to our recorded show notes. They all have clickable links that will take you directly to that part of the audio.

Previous podcasts

Haley Tucker on Responding to Failures in Playback Features at Netflix

Kolton Andrus on Lessons Learnt from Failure Testing at Amazon and Netflix and New Venture Gremlin

Preslav Le on How Dropbox Moved off AWS and What They Have Been Able to Do Since

Randy Shoup on Stitch Fix's Technology Stack, Data Science and Microservices

Personas Architecture & Design Topics Database Slack SQL MySQL InfoQ Relational Databases The InfoQ Podcast

Related Editorial

- Uber Engineering Moving from Postgres to MySQL
- Elasticsearch for SQL Users
- Federated Queries with HAWQ - SQL on Hadoop and Beyond
- How Slack Works

But I Need a Database that _Scales_

Tell us what you think

Please enter a subject

Message

Post Message

Community comments

Watch Thread

InfoQ Weekly Newsletter

Subscribe to our Weekly email newsletter to follow all new content on InfoQ

Click to view how it looks like

Your email here Subscribe

Development

Visual Studio Code 1.8 Adds Hot Exit, Zen Mode, More Debugging Options

Swift and Objective-C Runtime Programming

Google Targets IoT with Android Things

Architecture & Design

Keith Adams on the Architecture of Slack, using MySql, Edge Caching, & the backend Messaging Server

Google Targets IoT with Android Things

Stranger Things: The Forces that Disrupt Netflix

Culture & Methods

Using a Skill Matrix for Growth and Learning

Key Takeaway Points and Lessons Learned from QCon San Francisco 2016

Why Agile Works

Data Science

Julien Nioche on StormCrawler, Open-Source Crawler Pipelines Backed by Apache Storm

Key Takeaway Points and Lessons Learned from QCon San Francisco 2016

Facebook Builds an Efficient Neural Network Model over a Billion Words

DevOps

Modern Web Security, Lazy but Mindful Like a Fox

Spinnaker - Land of a 1000 Builds

Key Takeaway Points and Lessons Learned from QCon San Francisco 2016

QCONS WORLDWIDE

- London Mar 6-10, 2017
- Beijing Apr 16-18, 2017
- São Paulo Apr 24-26, 2017
- New York Jun 26-30, 2017
- Shanghai Oct 19-21, 2017
- Tokyo Fall, 2017
- San Francisco Nov 13-17, 2017

InfoQ Weekly Newsletter

Subscribe to our Weekly email newsletter to follow all new content on InfoQ

Click to view an example

Your email here Subscribe

- **Your personalized RSS**
- **For daily content and announcements**
- **For major community updates**
- **For weekly community updates**

Personalize Your Main Interests

- [x] **Development**
- [x] **Architecture & Design**
- [x] **Data Science**
- [x] **Culture & Methods**
- [x] **DevOps**

This affects what content you see on the homepage & your RSS feed. Click preferences to access more fine-grained personalization.

General Feedback feedback@infoq.com

Bugs bugs@infoq.com

Advertising sales@infoq.com

Editorial editors@infoq.com

Marketing marketing@infoq.com