

FINAL PROJECT

present

Scalability of Twitter

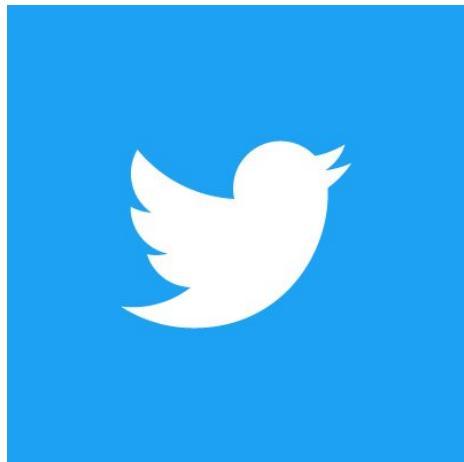
โดย

นางสาว สิริญาพร นิลกรรณ์ เลขประจำตัว 5809700023

เสนอ

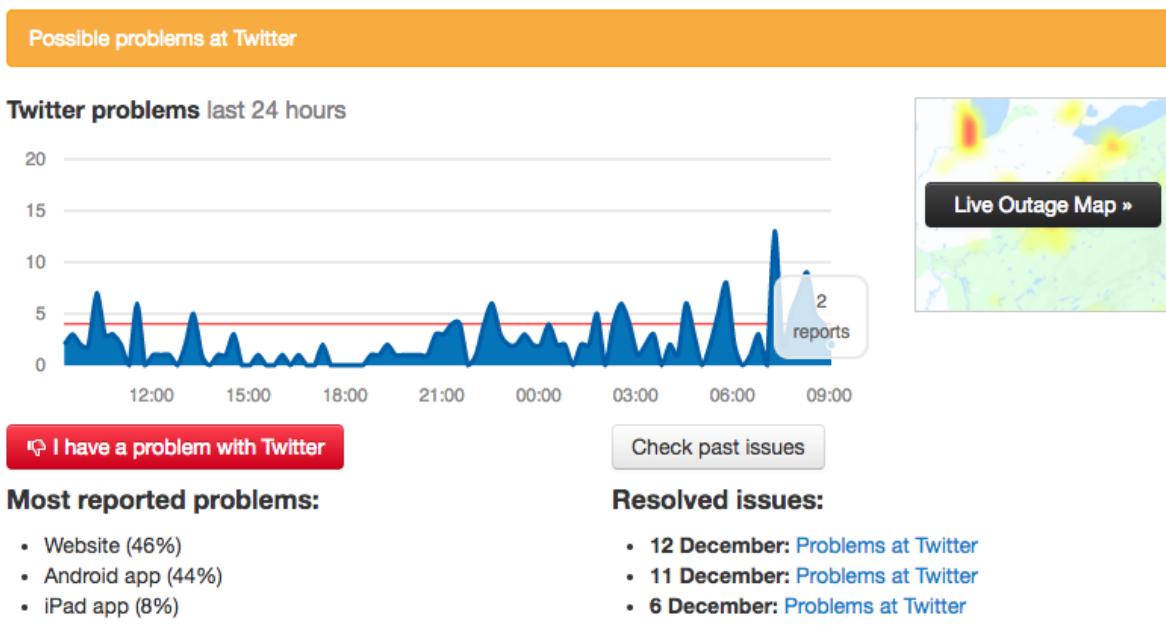
ดร.ประภาพร รัตนธำรง

วิชาระบบปฏิบัติการ 2 คพ.447
คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยธรรมศาสตร์



Twitter เป็นบริการเครือข่ายสังคมออนไลน์จำพวก Micro Block ทวิตเตอร์ก่อตั้งโดยบริษัท Obvious Corp สร้างข้อความที่เราต้องการลงบนหน้า PROFILE เช่น ว่าตอนนี้กำลังทำอะไรอยู่ รู๊สิก อย่างไร โดยผู้ใช้สามารถส่งข้อความยาวไม่เกิน 140 ตัวอักษร และสามารถแจ้งข่าวทางธุรกิจบนโลก สังคมออนไลน์ได้ เช่น ถ้าหากว่าบริษัทกำลังมีข่าวอะไรจะอัพเดตกับคนทั่วไป ก็ใช้ Twitter แจ้งข่าว ได้ถ้าคนที่สนใจในสินค้าและบริการของเรา เขารายกติดตามเรารอยู่แล้ว การแจ้งข่าวแบบนี้ควรทำให้บ่อย เพื่อให้สมาชิก Twitter รู้สึกได้ว่าคุณมีความเคลื่อนไหว โดยการส่งข้อความนี้ว่า ทวีต(อังกฤษ:Tweet) ซึ่งแปลว่า เสียงกรอง ได้รับความนิยมจากทั่วโลก โดยผู้ใช้ที่ลงทะเบียนเข้าใช้มากกว่า 500 ล้านคน ในปี พ.ศ. 2555 รวมไปถึงมีทวีตมากกว่า 340 ล้านทวีตต่อวัน และมีการค้นหา ข้อมูลผ่านเว็บไซต์มากกว่า 1,600 ล้านครั้งต่อวัน นอกจากที่จะสามารถทวีตบนเว็บไซต์แล้ว ได้มีการ เปิดให้ใช้งานการส่งทวีต ด้วยการส่งบริการข้อความสั้น (SMS) และบนโปรแกรมประยุกต์ ใน โทรศัพท์มือถือ และสมาร์ต โฟน ตัวระบบซอฟต์แวร์ของทวิตเตอร์ เดิมพัฒนาด้วยรูบีอนเรลส์จนเมื่อ ราวสิ้นปี ค.ศ. 2008 หรือ พุทธศัลกราช 2551 รีนั่นเอง จึงได้เปลี่ยนมาใช้ภาษา Scala บนแพลตฟอร์ม จาวา ข้อความอัปเดตที่ส่งเข้าไปยังทวิตเตอร์จะแสดงอยู่บนเว็บเพจของผู้ใช้คน นั้นบนเว็บไซต์ และผู้ใช้คนอื่นสามารถเลือกรับข้อความเหล่านี้ทางเว็บไซต์ทวิตเตอร์ อีเมล เอสเอ็มเอส เมสเซนเจอร์ (IM), RSS, หรือผ่านโปรแกรมเฉพาะอย่าง Twitterific Twhirl ปัจจุบันทวิตเตอร์มีหมายเลขอโทรศัพท์ สำหรับส่งเอสเอ็มเอส

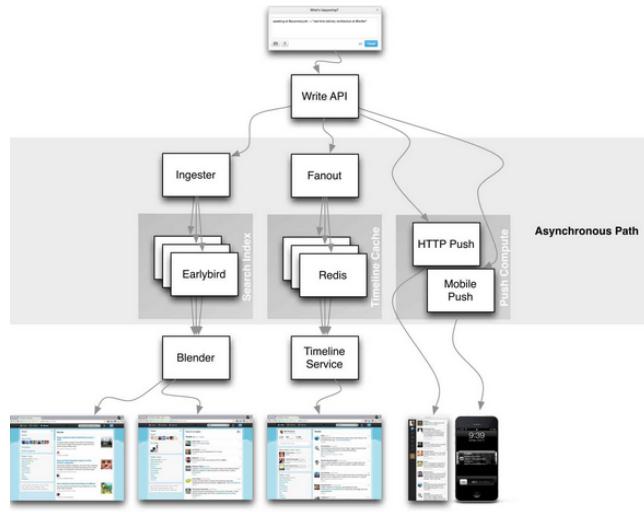
ประเด็นปัญหาเกี่ยวกับ Scalability ที่ประสบในการให้บริการ



จากปัญหาดังกล่าวข้อมูล

มีปัญหาระบบจำนวนมากขึ้น โปรแกรมการเข้าชมมีการเปลี่ยนแปลงมากขึ้นทำให้ช้าและเก็บข้อมูลได้ไม่เพียงพอต่อผู้ใช้ และไม่ก่อปิดต่อมาก่อนยื่นข้อมูลทวิตเตอร์มีขนาดใหญ่กว่าเดิม 400% มีสมมติฐานเดิมที่แนะนำการออกแบบเครือข่ายในระยะแรกไม่ประสบผลลัพธ์เจ้าที่ควร การเข้าชมของผู้ใช้เดิบโตขึ้นเร็วกว่าที่สามารถสร้างสถาปัตยกรรมใหม่ให้กับดาต้าเซ็นเตอร์ได้ ดังนั้น หากพิจารณาจำนวนทรีต์ที่ต้องได้รับการประมวลผล จัดเก็บ และเรียกดันจะสามารถดูได้ว่าการขยายขนาดได้อย่างรวดเร็วและเป็นอย่างไร

เทคนิคไวรี หรือสถาปัตยกรรมที่ใช้ในการแก้ปัญหา

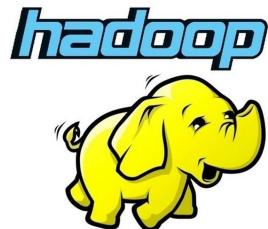


เป็นเรื่องสำคัญที่จะต้องสร้างสถาปัตยกรรมแบบปรับขนาดได้สูงซึ่งจะช่วยเพิ่มขีดความสามารถเพิ่มขึ้นแทนในการโยกย้ายแบบ microservices fanout สูง ต้องการเครือข่ายความนำเชื่อถือสูงที่สามารถจัดการกับความหลากหลายของการเข้าชม ดังนั้น มีการออกแบบในภายหลัง

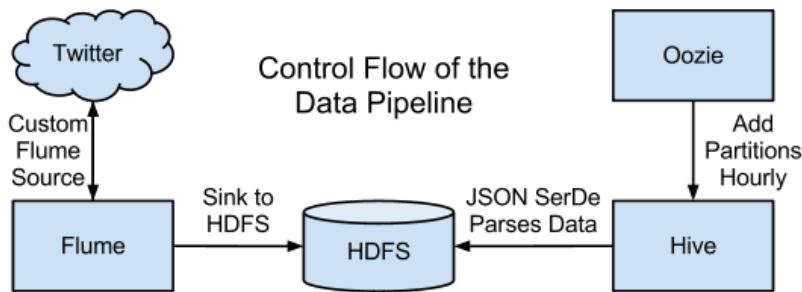
1. โดยใช้ CND เป็นบัฟเฟอร์ขนาดมาตรฐานส่งผ่าน HTTP request และแบ่งพาร์ติชันโดยแคชของเราโดยผู้ใช้ทวีตใหม่เป็นต้น โดยทั่วไปแคชทุกกลุ่มจะถูกปรับตามความต้องการโดยเฉพาะ ขึ้นอยู่กับชนิดของคลัสเตอร์พวกรเข้าจัดการระหว่าง 10M ถึง 50M QPS และทำงานระหว่างหلامร้อยละ พันกรณี



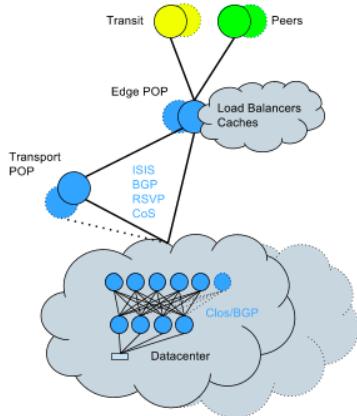
2. เราสามารถใช้ hadoop ใน การนำส่วนประกอบบางส่วนของ CDH มารวมกันเพื่อสร้างระบบการจัดการข้อมูลแบบ end-to-end สถาปัตยกรรมแบบเดียวกันนี้สามารถใช้กับแอปพลิเคชันที่หลากหลาย



ออกแบบมาเพื่อดึงข้อมูล Twitter เช่นการระบุบัญชีสแปมหรือการระบุกลุ่มคำหลัก สถาปัตยกรรมทั่วไปสามารถใช้งานได้หลายรูปแบบ ด้วยการเชื่อมต่อแหล่ง Flume ต่างๆ และ Hive SerDes เพื่อให้เรียกค้น จัดเก็บได้อย่างรวดเร็ว



3. แปลงศูนย์ข้อมูลที่มีอยู่แล้วเป็นโครงสร้างแบบ Clos + BGP ซึ่งการแปลงที่ต้องดำเนินการบนเครือข่ายออนไลน์ยังทำได้แม้จะมีความซับซ้อนเสร็จสิ้นแล้ว แต่ก็มีผลกระทบน้อยที่สุดต่อบริการในช่วงเวลาอันสั้น ขณะนี้เครือข่ายมีลักษณะดังนี้:



3.1. ทำให้เล็กกว่าอุปกรณ์ Smaller blast radius of a single device failure.

3.2. ใช้ความสามารถในการปรับขนาดแบนด์วิดท์ตามแนวอน (Horizontal bandwidth scaling capabilities)

3.3. ลดค่าใช้จ่ายของชีพียูในการกำหนดเส้นทาง มีประสิทธิภาพมากขึ้นในการประมวลผลการปรับปรุงเส้นทาง

(Lower routing engine CPU overhead; far more efficient processing of route updates.)

3.4. ทำความจุเส้นทางที่สูงขึ้นเนื่องจากการลดค่าใช้จ่ายของ CPU

(Higher route capacity due to lower CPU overhead.)

3.5. ควบคุมนโยบายการกำหนดเส้นทางแบบละเอียดมากขึ้น ในแต่ละอุปกรณ์และพื้นฐานการเข้ามายิงเพิ่มระยะเวลา reconvergence โพรโทคอลเส้นทางปัญหา churn และปัญหาที่ไม่คาดคิดกับ OSPF ความซับซ้อนโดยธรรมชาติช่วยให้สามารถโยกย้ายชั้นที่ไม่มีผลกระทบ

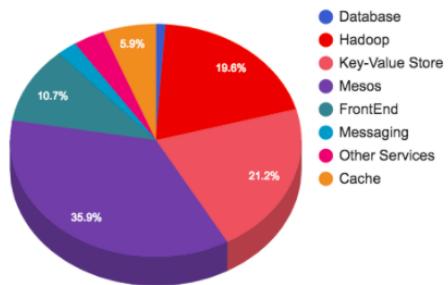
(More granular control of routing policy on a per device and link basis.

No longer exposed to several known root causes of prior major incidents:

increased protocol reconvergence times, route churn issues and unexpected issues with inherent OSPF complexities. Enables non-impacting rack migrations.)

ผลที่ได้เมื่อใช้เทคนิคชี้ดังกล่าว

Our current distribution of hardware is shown below:

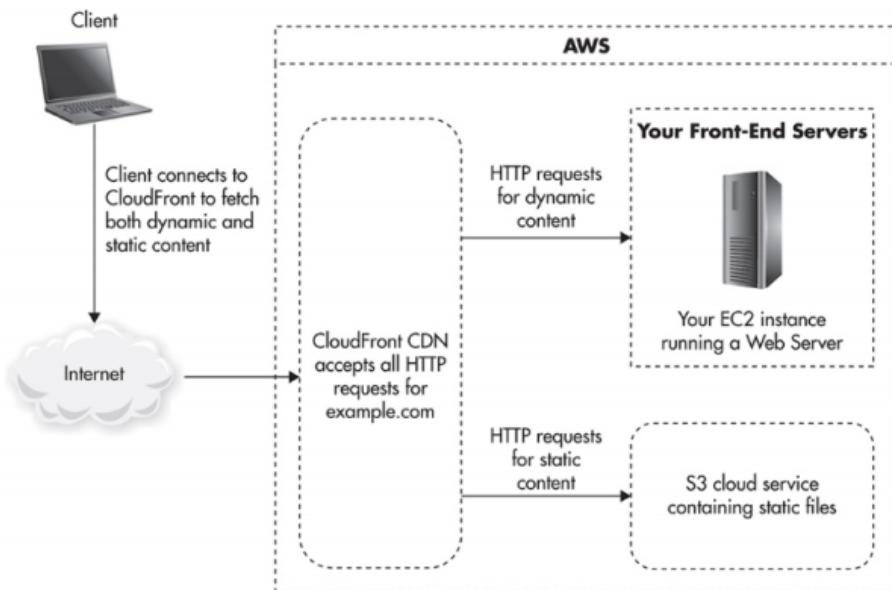


- จำนวนทวีตที่ต้องได้รับการประมวลผล จัดเก็บและเรียกคืนจะสามารถดูได้ว่าการขยายขนาดได้อย่างรวดเร็ว
- สามารถดูว่าผู้ที่มีผู้ชมมากที่สุดจะได้เห็นผู้คนที่ทวีตหรือไม่และยังกำหนดว่าคนเหล่านี้กำลังสื่อสารอยู่เป็นประจำหรือไม่ เราสามารถใช้ข้อมูลนี้เพื่อกำหนดเป้าหมายการรับส่งข้อความของเราอย่างรอบคอบ
- ใช้ข้อมูลและเมตริกในการตัดสินใจทางด้านเทคนิคที่ถูกต้องและมั่นใจได้ว่าชีวิตเหล่านี้สามารถเข้าใจได้กับผู้ดำเนินการเครือข่ายซึ่งมีความสำคัญเป็นอย่างยิ่ง ในสภาพแวดล้อมแบบไฮสต์และแบบคลาวด์
- ส่วนน้อยในคนส่วนใหญ่ที่ใช้วิธีแก้ปัญหาในเทคโนโลยีนี้

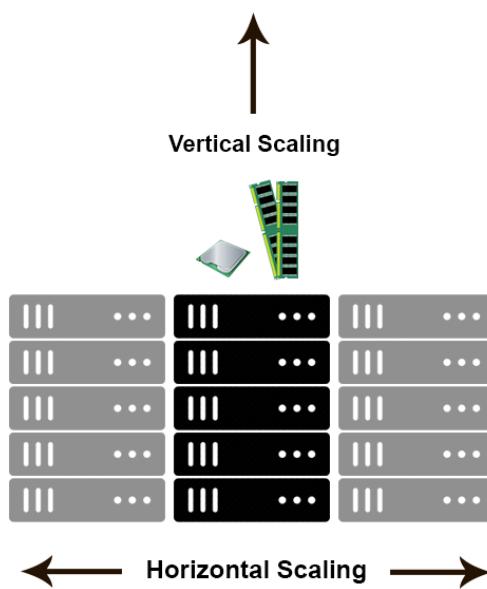


อภิปรายความสัมพันธ์กับเนื้อหาที่เรียนในชม.บรรยาย

1. ใช้ CDN (Cloud Delivery Network) ในการเก็บข้อมูล Cache โดยใช้ HTTP request

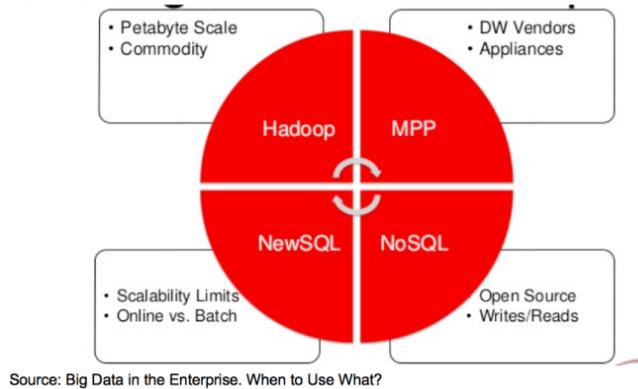


2. มีการทำ Horizontal bandwidth scaling capabilities



3. ใช้ Hadoop ส่วนประกอบคล้ายกับ nosql ที่ได้ศึกษาในห้องเรียน

Big Data Landscape



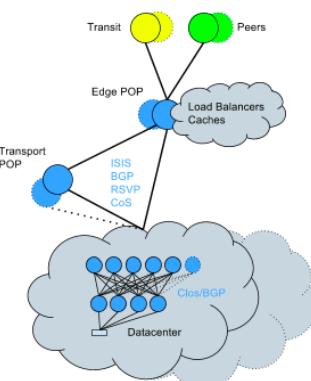
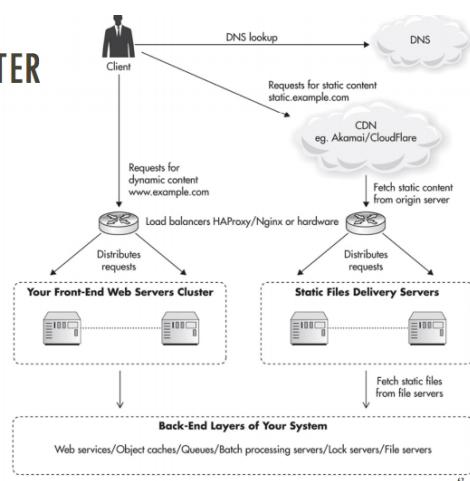
IN-MEMORY KEY VALUE STORE

- An in-memory key-value store is a **NoSQL database** optimized for **read-heavy** application workloads (such as social networking, gaming, media sharing and Q&A portals) or **compute-intensive** workloads (such as a recommendation engine).
- Memcached: a widely adopted memory object caching system.
 -
- Redis: a widely adopted memory object caching system.
 - <https://codeforgeek.com/2015/07/using-redis-to-handle-session-in-node-js/>
 - <https://slaptijack.com/system-administration/using-redis-for-php-session-storage.html>

<https://slaptijack.com/system-administration/using-redis-for-php-session-storage.html>

4. มีการใช้เทคนิคหลักคือ Deploy data center

PRIVATE DATA CENTER DEPLOYMENT



แหล่งอ้างอิงที่น่าเชื่อถือ



- <https://developer.twitter.com/#>

Twitter คืออะไร มีประโยชน์อย่างไร มีกี่แบบ และมีไว้เพื่ออะไร?

- <https://th.answers.yahoo.com/question/index?qid=20090512201700AAi7ph9>

The Architecture Twitter Uses To Deal With 150M Active Users, 300K QPS, A 22 MB/S Firehose, And Send Tweets In Under 5Seconds

- <http://highscalability.com/blog/2013/7/8/the-architecture-twitter-uses-to-deal-with-150m-active-users.html>

How to answer design question: How do you design a twitter?

- <https://www.youtube.com/watch?v=gX8S7b8UYI8>

The Infrastructure Behind Twitter: Scale

- https://blog.twitter.com/engineering/en_us/topics/infrastructure/2017/the-infrastructure-behind-twitter-scale.html

