



Study of Real-Time Data Communication using Pulsar and RabbitMQ (Case Study of Stock Price in Lao Securities Exchange)

Adsavin THEPPHAKAN, Vimontha KHEOVONGPHACHANH

Sokpaluang village, Faculty of Engineering, National University of Laos

Abstract

***Correspondence:** Adsavin
THEPPHAKAN, Sokpaluang
village, Lao Thai road, Faculty of
Engineering, National University of
Laos, telephone: +8562022224071,
email: adsavin@live.com

Article Info:

Submitted: February 06, 2025

Revised: March 30, 2025

Accepted: April 09, 2025

This study aims to analyze the performance speed of data transmission between Pulsar and RabbitMQ, compare their efficiency in handling different message sizes ranging from 1 KB to 1,000 KB, and evaluate the suitability of each system for real-time data transmission in the Lao Securities Market. The testing was conducted in a Docker environment on MacOS Air M2 using NodeJS as the development tool. The experiment simulated real-time stock price data transmission in the Lao Securities Market, testing with progressively increasing message sizes.

The results showed that RabbitMQ performed faster with small-sized messages (1 - 10 KB), taking only 3 - 4.44 milliseconds compared to Pulsar's 16 - 17.7 milliseconds. However, when message sizes increased to 500 - 1,000 KB, Pulsar demonstrated superior performance maintenance, requiring only 19 - 26 milliseconds, while RabbitMQ's processing time significantly increased to 54 - 95 milliseconds.

These findings indicate that Pulsar is more suitable and efficient than RabbitMQ for securities trading systems that require handling larger message sizes and maintaining stable performance across varying data volumes. Meanwhile, RabbitMQ might be more appropriate for the high-speed transmission of smaller messages. The study's results provide valuable insights for selecting the appropriate messaging system based on specific use cases in the Lao Securities Market.

Keywords: Message Broker, Message Queue, Pulsar, RabbitMQ, Real-Time.

1. ພາກສະເໜີ

ໃນຍຸກແຫ່ງການພັດທະນາທາງດ້ານດິຈິຕອນ, ການຈັດການຂໍ້ມູນແບບ Real-time ກາຍເປັນອົງປະກອບສໍາຄັນໃນການພັດທະນາລະບົບ Software ສະໄໝໃໝ່ (Sundar Rajan et al., 2021). ໂດຍສະເພາະໃນຂົງເຂດຕະຫຼາດການເງິນ ແລະ ຕະຫຼາດຫຼັກຊັບ, ປະສິດທິພາບດ້ານຄວາມໄວໃນການຮັບ-ສົ່ງຂໍ້ມູນມີຄວາມສໍາຄັນຢ່າງຍິ່ງຕໍ່ການຕັດສິນໃຈຂອງນັກລົງທຶນ (Mehendale, 2018).

ເນື່ອງຈາກຂໍ້ມູນຕ່າງໆ ໃນລະບົບຂອງຕະຫຼາດຫຼັກຊັບອາດມີການປ່ຽນແປງຢ່າງວ່ອງໄວ ຖ້າວ່າ ລະບົບ ຫຼື ເຕັກໂນໂລຊີ ທີ່ນໍາມາໃຊ້ໃນການກະຈາຍຂໍ້ມູນໄປສູ່ລະບົບພາຍນອກຫາກຫຼ້າຊ້າ ອາດນໍາໄປສູ່ການສູນເສຍໂອກາດໃນການລົງທຶນ, ເພາະວ່າການຕັດສິນໃຈລົງທຶນໃນຫຼັກຊັບຂອງນັກລົງທຶນນັ້ນຄືຕ້ອງການກໍາໄລຈາກສ່ວນຕ່າງໃນການຊື້ຂາຍຫຼັກຊັບ (LATHILATH et al., 2020).

Pulsar ເປັນ Message Broker ທີ່ໄດ້ຮັບການພັດທະນາດ້ານຄວາມສາມາດໃນການຂະຫຍາຍຕົວ ໂດຍມີບໍລິສັດຫຼາກຫຼາຍຂະໜາດ

ຫຼາຍຮ້ອຍແຫ່ງນໍາໃຊ້, ສາມາດຮອງຮັບການສົ່ງຂໍ້ຄວາມໄດ້ຫຼາຍລ້ານຂໍ້ຄວາມຕໍ່ວິນາທີ (Apache Pulsar, 2025), ເຊິ່ງຖືກພັດທະນາໂດຍບໍລິສັດ Yahoo ແລະ ເປີດຕົວໃນປີ 2016, ໂດຍໄດ້ນໍາສະເໜີແນວທາງໃໝ່ໃນການຈັດການຂໍ້ມູນແບບ Distributed Streaming (Andström, 2024).

RabbitMQ ເປັນ Message Broker ທີ່ໄດ້ຮັບການຍອມຮັບຢ່າງກວ້າງຂວາງໃນການຈັດການຮັບ-ສົ່ງຂໍ້ມູນແບບ Real-time, ການຕິດຕັ້ງ ແລະ deploy ມີຄວາມງ່າຍ ໃນຫຼາກຫຼາຍສະພາບແວດລ້ອມ (RabbitMQ, 2025), ເປັນ Open source ທີ່ມີມາແຕ່ປີ 2007, ໄດ້ພັດທະນາໜ້າເຊື່ອຖືຂອງຕົນໃນອຸດສາຫະກຳຫຼາຍປະເພດ (Rostanski et al., 2014).

ການພັດທະນາແບບ Containerization ໂດຍສະເພາະການນໍາໃຊ້ Docker ໄດ້ປ່ຽນແປງວິທີການພັດທະນາ ແລະ ການ Deploy Software (Rad et al., 2016) ໃຫ້ມີຄວາມງ່າຍດາຍ ແລະ ສະດວກຍິ່ງຂຶ້ນ ເຊິ່ງເຮັດໃຫ້ເໝາະສົມສໍາລັບການທົດສອບປະສິດທິພາບຂອງ

ລະບົບ ໂດຍທີ່ບໍ່ຈຳເປັນຕ້ອງລົງທຶນຫຼາຍໃນການຊື້ອຸປະກອນ Hardware.

Node.js ເປັນພາສາໂປຣແກຣມ ທີ່ມີຄວາມໂດດເດັ່ນທາງດ້ານ Event-driven ແລະ non-blocking I/O Model, ເປັນເຕັກໂນໂລຊີທີ່ເໝາະສົມສຳລັບການພັດທະນາແອັບພິກເຄຊັນເວລາຈິງ (Real-time Application) ການເຮັດວຽກຮ່ວມກັບ Message Queue ໃນການຮັບ-ສົ່ງຂໍ້ມູນຜ່ານ Node.js ສາມາດສະແດງໃຫ້ເຫັນປະສິດທິພາບທີ່ແທ້ຈິງໃນການຈັດການກັບຂໍ້ມູນຂະໜາດຕ່າງໆ.

ຕະຫຼາດຫຼັກຊັບລາວ (LSX) ໄດ້ເລີ່ມດຳເນີນການໃນວັນທີ 10 ຕຸລາ 2010, ເປັນຜົນມາຈາກການຮ່ວມມືລະຫວ່າງລັດຖະບານລາວ ແລະ ຕະຫຼາດຫຼັກຊັບ ສ.ເກົາຫຼີ. ນັບແຕ່ມີການສ້າງຕັ້ງເປັນຕົ້ນມາ LSX ໄດ້ເຕີບໂຕຢ່າງຕໍ່ເນື່ອງ ໂດຍສະເພາະແມ່ນຈຳນວນບໍລິສັດຈິດທະບຽນໃນປີ 2010 ມີພຽງ 2 ບໍລິສັດ ແຕ່ໃນປະຈຸບັນ ໄດ້ເພີ່ມຂຶ້ນເປັນ 10 ບໍລິສັດ (LATHILATH et al., 2020). ການເພີ່ມຂຶ້ນຂອງປະລິມານການຊື້-ຂາຍ ແລະ ຄວາມຕ້ອງການໃນການເຂົ້າເຖິງຂໍ້ມູນແບບທັນທີ ທັນໃດຈາກນັກລົງທຶນ ເຮັດໃຫ້ການສຶກສາປະສິດທິພາບຂອງລະບົບ Message Queue ມີຄວາມສຳຄັນຫຼາຍຂຶ້ນ.

ສະນັ້ນ, ການສຶກສານີ້ມີຈຸດປະສົງເພື່ອ: ວິເຄາະປະສິດທິພາບດ້ານຄວາມໄວໃນການຮັບ-ສົ່ງຂໍ້ມູນຂອງ Pulsar ແລະ RabbitMQ, ປຽບທຽບປະສິດທິພາບໃນການຮັບ-ສົ່ງຂໍ້ມູນໃນຂະໜາດຕ່າງໆ (1 KB ຫາ 1,000 KB) ແລະ ປະເມີນຄວາມເໝາະສົມຂອງແຕ່ລະລະບົບສຳລັບການໃຊ້ງານໃນການຮັບ-ສົ່ງຂໍ້ມູນລາຄາຮຸ້ນໃນຕະຫຼາດຫຼັກຊັບລາວ.

ວິທີການສຶກສາຈະດຳເນີນການທົດລອງໃນສະພາບແວດລ້ອມທີ່ຄວບຄຸມໄດ້, ໂດຍມີການໃຊ້ MacBook Air M2 ເປັນຄອມພິວເຕີ

ຈຳລອງໃນການທົດສອບ. ການທົດລອງຈະໃຊ້ Node.js ເພື່ອພັດທະນາແອັບພິກເຄຊັນ Producer ແລະ Consumer, ເຊິ່ງຈະດຳເນີນການທົດສອບການສົ່ງຂໍ້ມູນຂະໜາດຕ່າງໆ ແລະ ຈະມີການວັດແທກເວລາທີ່ໃຊ້ໃນການສົ່ງ-ຮັບຂໍ້ມູນໃນຫຼາກຫຼາຍຂະໜາດ ຈາກນັ້ນອາດໄປຫາໃຫຍ່ແຕກຕ່າງກັນ.

ຄວາມສຳຄັນຂອງຜົນການສຶກສານີ້ຈະເປັນປະໂຫຍດບໍ່ພຽງແຕ່ສຳລັບຕະຫຼາດຫຼັກຊັບລາວເທົ່ານັ້ນ, ແຕ່ຍັງສາມາດນຳໄປປະຍຸກໃຊ້ກັບຕະຫຼາດຫຼັກຊັບອື່ນໆໃນພາກພື້ນທີ່ມີຂະໜາດ ແລະ ຄວາມຕ້ອງການທີ່ຄ້າຍຄືກັນ. ນອກຈາກນັ້ນ ຍັງສາມາດໄປປະຍຸກໃຊ້ໃນການຮັບ-ສົ່ງຂໍ້ມູນອື່ນໆ ທີ່ຕ້ອງການການຮັບ-ສົ່ງຂໍ້ມູນແບບ Real-time ເຊັ່ນ: ລະບົບຄຳຂາຍອີເລັກໂທຣນິກ (E-commerce) ທີ່ຕ້ອງຈັດການກັບຂໍ້ມູນລາຄາແບບທັນທີ, ຂໍ້ມູນການສັ່ງຊື້ ແລະ ການຊຳລະເງິນແບບດິຈິຕອນທີ່ຕ້ອງການການປະມວນຜົນທຸລະກຳທີ່ໄວ ແລະ ຖືກຕ້ອງ (Ureche et al., 2000), ລະບົບ IoT (Internet of Things) ທີ່ຕ້ອງການການເກັບກຳ ແລະ ວິເຄາະຂໍ້ມູນຈາກອຸປະກອນຈຳນວນຫຼາຍໃນເວລາດຽວກັນ (Toprak et al., 2021)

2. ອຸປະກອນ ແລະ ວິທີການ

2.1 ຂໍ້ມູນໃນການທົດລອງ

ໃນການທົດລອງຮັບ-ສົ່ງຂໍ້ມູນຄັ້ງນີ້ ແມ່ນຈະໄດ້ຈຳລອງການຮັບຂໍ້ມູນຈາກຕະຫຼາດຫຼັກຊັບລາວ ຜ່ານ UDP (User Datagram Protocol) (ດັ່ງຮູບພາບທີ 1) ແລະ ນຳໃຊ້ຂໍ້ມູນຈຳລອງລາຄາຮຸ້ນຂອງຕະຫຼາດຫຼັກຊັບລາວ ເຊິ່ງຈະມີການກຳນົດຄວາມຍາວ, ຕຳແໜ່ງ ແລະ ປະເພດຂອງຂໍ້ມູນ ດັ່ງຕາຕະລາງທີ 1 ລຸ່ມນີ້:

ຕາຕະລາງທີ 1: ໂຄງສ້າງຂໍ້ມູນຈຳລອງລາຄາຮຸ້ນຂອງບໍລິສັດໃນຕະຫຼາດຫຼັກຊັບລາວ

ລ/ດ	ພາກສ່ວນຂໍ້ມູນ	ປະເພດຂໍ້ມູນ	ຄວາມຍາວ	ຕຳແໜ່ງ	ຄວາມຍາວລວມ
1	ລະຫັດປະເພດຂໍ້ຄວາມ	String	10	0-10	10
2	ວັນທີເວລາຂໍ້ມູນ	String	14	10-24	24
3	ປະເພດຫຼັກຊັບ	String	10	24-34	34
4	ລະຫັດດັດສະນີ	String	20	34-54	54
5	ລະຫັດຮຸ້ນ (ບໍລິສັດ)	String	30	54-84	84
6	ຊື່ຮຸ້ນ (ບໍລິສັດ)	String	500	84-584	584
7	ລາຄາປະຈຸບັນ	Number	20	584-604	604
8	Indicator sign (+/-)	String	1	604-605	605
9	ລາຄາປ່ຽນແປງ	Number	20	605-625	625
10	ຈຳນວນຮຸ້ນທີ່ຊື້ຂາຍ	Number	20	625-645	645
11	ມູນຄ່າການຊື້ຂາຍ	Number	20	645-665	665
12	ຈຳນວນຮຸ້ນທັງໝົດ	Number	20	665-685	685

13	ລາຄາ Par	Number	20	685-700	700
14	ລະຫັດຜູ້ສົ່ງ	String	50	700-750	750
15	ລະຫັດຜູ້ຮັບ	String	50	750-800	800
16	Check Sum	String	200	800-1,000	1,000 (ທຽບເທົ່າ 1 KB)

2.2 ເຄື່ອງມືໃນການທົດລອງ
ການທົດລອງຄັ້ງນີ້ແມ່ນໄດ້ນຳໃຊ້ ເຄື່ອງມື ທີ່ເປັນທັງ
Software ແລະ Hardware ຄືດັ່ງນີ້:

- 1) ເຄື່ອງມືທີ່ເປັນ Software:
 - (1) Visual Code (version 1.92.0) ເປັນເຄື່ອງມື Software ທີ່ໃຊ້ໃນການຂຽນ ແລະ ແລ່ນຄຳສັ່ງ.
 - (2) NodeJS (version 22.6.0) ເປັນພາສາໂປຣແກຣມ (Programming Language) ໃນການຂຽນຄຳສັ່ງ (Code) ໃນການຮັບ-ສົ່ງຂໍ້ມູນ ຜ່ານທັງ 2 ໂຕກະຈາຍຂໍ້ມູນ (Pulsar ແລະ RabbitMQ) ເຊິ່ງຈະເປັນການຂຽນ Code ຂຶ້ນມາໃໝ່ສະເພາະຝັງຊັນ (Function) ທີ່ຈະໃຊ້ໃນການທົດລອງເທົ່ານັ້ນ ເຊັ່ນ: ການຮັບ-ສົ່ງຂໍ້ມູນ Text ຈາກ 1 ຜູ້ສົ່ງ ຫາ 1 ຜູ້ຮັບ.
 - (3) Docker (version 4.35.1) ເປັນ ສະພາບແວດລ້ອມຂອງເຊີເວີ ທີ່ຮອງຮັບການຕິດຕັ້ງ Software ກະຈາຍຂໍ້ມູນ ທັງ Pulsar ແລະ RabbitMQ.
 - (4) Pulsar (version 4.0) ເປັນ Software ກະຈາຍຂໍ້ມູນ
 - (5) RabbitMQ (version 3.13) ເປັນ Software ກະຈາຍຂໍ້ມູນ
- 2) ເຄື່ອງມືທີ່ເປັນ Hardware:
 - (1) ເຄື່ອງຄອມພິວເຕີ Macbook Air ທີ່ມີ RAM 24 GB ແລະ CPU M2 ສະຖາປັດຕະຍະກຳ ARM ຂອງ Apple Silicon ທີ່ເຮັດວຽກພາຍໃຕ້ລະບົບປະຕິບັດການ macOS Sonoma 14.5.

2.3 ການຄວບຄຸມຕົວປ່ຽນ
ໃນການທົດລອງຄັ້ງນີ້, ຜູ້ຄົນຄວ້າໄດ້ມີການຄວບຄຸມຕົວປ່ຽນທີ່ອາດເປັນປັດໄຈພາຍນອກທີ່ມີຜົນເຮັດໃຫ້ການທົດລອງບໍ່ຊັດເຈນດັ່ງລຸ່ມນີ້:

- ປັດການນຳໃຊ້ແອັບພລິເຄຊັນອື່ນໆທີ່ບໍ່ກ່ຽວຂ້ອງໃນລະຫວ່າງການທົດສອບເພື່ອຮັກສາຊັບພະຍາກອນຂອງເຄື່ອງຄອມພິວເຕີ ເຊັ່ນ: CPU, RAM ໃຫ້ມີຄວາມຄົງທີ່ທີ່ສຸດ.
- ປັດການເຊື່ອມຕໍ່ເຄືອຂ່າຍຕ່າງໆ ເຊັ່ນ: LAN, Wi-fi, Bluetooth...
- ເຊື່ອມຕໍ່ເຄື່ອງຄອມພິວເຕີກັບສາຍສາກຕະຫຼອດຊ່ວງເວລາຂອງການທົດລອງ ເພື່ອໃຫ້ປະສິດທິພາບຂອງເຄື່ອງຖືກໃຊ້ຢ່າງເຕັມທີ່.

2.4 ຮູບແບບການທົດລອງ

ໃນການທົດລອງຄັ້ງນີ້ແມ່ນໄດ້ມີການແບ່ງຂະໜາດຂໍ້ມູນທີ່ໃຊ້ໃນການທົດສອບອອກເປັນ 5 ຂະໜາດຂໍ້ມູນດັ່ງນີ້: 1 KB, 10 KB, 100 KB, 500 KB ແລະ 1,000 KB, ເຊິ່ງໃນແຕ່ລະຂະໜາດຂໍ້ມູນຈະທຽບເທົ່າກັບຈຳນວນຂໍ້ມູນລາຄາຂອງຮຸ້ນຂອງບໍລິສັດ ໃນຕະຫຼາດຫຼັກຊັບລາວ ເຊັ່ນ: 1 KB ທຽບເທົ່າກັບການຮັບ-ສົ່ງຂໍ້ມູນລາຄາຮຸ້ນຂອງ 1 ບໍລິສັດ, 10 KB ທຽບເທົ່າກັບການຮັບ-ສົ່ງຂໍ້ມູນລາຄາຮຸ້ນຂອງ 10 ບໍລິສັດ, 100 KB ທຽບເທົ່າກັບການຮັບ-ສົ່ງຂໍ້ມູນລາຄາຮຸ້ນຂອງ 100 ບໍລິສັດ ... ຕາມລຳດັບ ເພື່ອໃຫ້ຮອງຮັບປະລິມານຂໍ້ມູນທີ່ມີຢູ່ໃນປະຈຸບັນ ແລະ ອາດຈະເພີ່ມຂຶ້ນໃນອະນາຄົດ.

2.5 ຂັ້ນຕອນການທົດສອບ

- 1) ຂຽນຄຳສັ່ງໂດຍນຳໃຊ້ ພາສາ Node.js ສ້າງ Producer ສົ່ງຂໍ້ມູນແບບຕໍ່ເນື່ອງ ໂດຍມີການແນບຄ່າຂອງເວລາຕອນສົ່ງເຂົ້າໄປໃນຂໍ້ຄວາມເພື່ອໃຫ້ຜູ້ຮັບເອົາໄປໃຊ້ຄິດໄລ່ຕໍ່ໄປ.
- 2) ຂຽນຄຳສັ່ງໂດຍນຳໃຊ້ ພາສາ Node.js ສ້າງ Consumer ຮັບຂໍ້ມູນ ແລະ ວັດແທກເວລາ ໂດຍໃຊ້ເວລາທີ່ແນບມາກັບຂໍ້ຄວາມ ລົບ ກັບ ເວລານະປະຈຸບັນທີ່ໄດ້ຮັບ ເພື່ອຫາສ່ວນຕ່າງ.
- 3) ທົດສອບການສົ່ງຂໍ້ມູນຈຳນວນ 10 ຂໍ້ຄວາມຕໍ່ 1 ຂະໜາດຂໍ້ມູນ ແລ້ວ ບັນທຶກໃສ່ໃນຕາຕະລາງ.
- 4) ວັດແທກຄວາມໄວສະເລ່ຍ ໂດຍຄິດໄລ່ຈາກເວລາທີ່ໃຊ້ໃນການສົ່ງ ແລະ ຮັບຂໍ້ມູນ ທັງໝົດ 10 ຂໍ້ຄວາມ ຕໍ່ 1 ຂະໜາດຂໍ້ມູນ.

2.6 ວິທີການບັນທຶກຜົນ

ໃນການທົດລອງສົ່ງ ແລະ ຮັບຂໍ້ມູນໃນແຕ່ລະຄັ້ງໃນແຕ່ລະຂະໜາດຂໍ້ມູນ ແມ່ນໄດ້ມີການບັນທຶກຜົນການທົດລອງລົງໃນ file ຮູບແບບ xls ສຳລັບ Microsoft Excel ເພື່ອຄິດໄລ່ຫາຄ່າສະເລ່ຍ (ຕາຕະລາງທີ 2) ແລະ ນຳເອົາຜົນການທົດລອງດັ່ງກ່າວໄປສ້າງເປັນເສັ້ນສະແດງ (ຮູບພາບທີ 2)

3. ຜົນໄດ້ຮັບ

- 1) ຜົນການວິເຄາະປະສິດທິພາບດ້ານຄວາມໄວໃນການຮັບ-ສົ່ງຂໍ້ມູນຂອງ Pulsar ແລະ RabbitMQ
 - ຈາກຜົນການທົດລອງພົບວ່າ RabbitMQ ໃຊ້ເວລາໃນການຮັບ-ສົ່ງຂໍ້ມູນຂະໜາດ 1 KB ພຽງແຕ່ 3 ມິນລີວິນາທີ, ໃນຂະນະທີ່ Pulsar ໃຊ້ເວລາ 16.48 ມິນລີວິນາທີ.
 - ການຮັບ-ສົ່ງຂໍ້ມູນ 10 KB, RabbitMQ ໃຊ້ເວລາ 4.44 ມິນລີວິນາທີ ແລະ Pulsar ໃຊ້ເວລາ 17.7 ມິນລີວິນາທີ.

- ການຮັບ-ສົ່ງຂໍ້ມູນຂະໜາດ 100 KB, RabbitMQ ໃຊ້ເວລາ 16.40 ມິນລິວິນາທີ ແລະ Pulsar ໃຊ້ເວລາ 18.94 ມິນລິວິນາທີ ເຊິ່ງເຫັນວ່າມີຄວາມແຕກຕ່າງກັນພຽງເລັກນ້ອຍ.
- 2) ຜົນການປຽບທຽບປະສິດທິພາບໃນການຮັບ-ສົ່ງຂໍ້ມູນຂະໜາດຕ່າງໆ (1 KB ຫາ 1,000 KB)
 - ສໍາລັບຂໍ້ມູນຂະໜາດນ້ອຍ (1 - 10 KB): RabbitMQ ມີປະສິດທິພາບດີກວ່າ Pulsar ຢ່າງຈະແຈ້ງ.
 - ສໍາລັບຂໍ້ມູນຂະໜາດກາງ (100 KB): ທັງສອງລະບົບມີປະສິດທິພາບໃກ້ຄຽງກັນ ໂດຍມີຄວາມແຕກຕ່າງພຽງ 2.54 ມິນລິວິນາທີ.
 - ສໍາລັບຂໍ້ມູນຂະໜາດໃຫຍ່ (500 - 1,000 KB): Pulsar ມີປະສິດທິພາບດີກວ່າ RabbitMQ ຢ່າງຊັດເຈນ. ສໍາລັບຂໍ້ມູນຂະໜາດ 500 KB, Pulsar ໃຊ້ເວລາ 19.26 ມິນລິວິນາທີ, ໃນຂະນະທີ່ RabbitMQ ໃຊ້ເວລາ 54.90 ມິນລິວິນາທີ. ສ່ວນຂໍ້ມູນຂະໜາດ 1,000 KB, Pulsar ໃຊ້ເວລາ 26.86 ມິນລິວິນາທີ ແຕ່ RabbitMQ ໃຊ້ເວລາເຖິງ 95.46 ມິນລິວິນາທີ.
- 3) ຜົນການປະເມີນຄວາມເໝາະສົມຂອງແຕ່ລະລະບົບສໍາລັບຕະຫຼາດຫຼັກຊັບລາວ
 - RabbitMQ ເໝາະສົມກັບການຮັບ-ສົ່ງຂໍ້ມູນຂະໜາດນ້ອຍ ເຊັ່ນ: ຂໍ້ມູນລາຄາຮຸ້ນຜື້ນຖານ, ການປ່ຽນແປງລາຄາ ແລະ ປະລິມານການຊື້-ຂາຍແບບ Real-time.
 - Pulsar ເໝາະສົມກັບການຮັບ-ສົ່ງຂໍ້ມູນຂະໜາດໃຫຍ່ ເຊັ່ນ: ຂໍ້ມູນການຊື້-ຂາຍປະຫວັດຍ້ອນຫຼັງ, ຂໍ້ມູນວິເຄາະຕະຫຼາດ ຫຼື ຂໍ້ມູນລາຍງານການເງິນຂອງບໍລິສັດຈິດທະບຽນ.
 - ການເລືອກໃຊ້ລະບົບໃດໜຶ່ງຄວນພິຈາລະນາຈາກປະເພດຂໍ້ມູນທີ່ມີການຮັບ-ສົ່ງເປັນສ່ວນໃຫຍ່ ແລະ ແຜນການຂະຫຍາຍຕົວໃນອະນາຄົດຂອງຕະຫຼາດຫຼັກຊັບລາວ.

4. ວິພາກຜົນ

- 1) ວິເຄາະປະສິດທິພາບດ້ານຄວາມໄວ:
 - RabbitMQ ມີປະສິດທິພາບດີກວ່າໃນການຮັບ-ສົ່ງຂໍ້ມູນຂະໜາດນ້ອຍ (1 - 10 KB) ໂດຍມີຄວາມໄວສູງກວ່າ Pulsar ປະມານ 4-5 ເທົ່າ.
 - Pulsar ມີປະສິດທິພາບດີກວ່າໃນການຮັບ-ສົ່ງຂໍ້ມູນຂະໜາດໃຫຍ່ (500 - 1,000 KB) ໂດຍມີຄວາມໄວສູງກວ່າ RabbitMQ 2-3 ເທົ່າ.
 - ທັງສອງລະບົບມີປະສິດທິພາບໃກ້ຄຽງກັນທີ່ຂະໜາດຂໍ້ມູນ 100 KB.
- ຜົນໄດ້ຮັບໃນການຄົ້ນຄວ້າທາງດ້ານຄວາມໄວໃນຄັ້ງນີ້ ແມ່ນມີຄວາມສອດຄ່ອງກັບການຄົ້ນຄວ້າຂອງ Chy et al. (2023) ທີ່ຜົບວ່າ Pulsar ມີປະສິດທິພາບສູງກວ່າໃນການຈັດການກັບຂໍ້ມູນຂະໜາດໃຫຍ່. ເຖິງຢ່າງໃດກໍຕາມ, ການສຶກສາຂອງ Dobbelaere & Esmaili (2017) ໄດ້ສະແດງໃຫ້ເຫັນວ່າ RabbitMQ ມີຄວາມໄດ້

ປຽບໃນການຈັດການກັບຂໍ້ມູນຂະໜາດນ້ອຍ, ເຊິ່ງມີຄວາມສອດຄ່ອງກັບຜົນການສຶກສາໃນຄັ້ງນີ້.

ຜົນການຄົ້ນຄວ້ານີ້ຍັງສອດຄ່ອງກັບທິດສະດີການຈັດການຂໍ້ມູນແບບກະຈາຍຂອງ John & Liu (2017) ທີ່ອະທິບາຍວ່າລະບົບທີ່ອອກແບບມາສໍາລັບການຮັບ-ສົ່ງຂໍ້ມູນຂະໜາດໃຫຍ່ມັກຈະໃຊ້ເວລາດົນ (Overhead) ໃນການຈັດການຮັບ-ສົ່ງຂໍ້ມູນຂະໜາດນ້ອຍ ແລະ ໃນທາງກົງກັນຂ້າມ ລະບົບຮັບ-ສົ່ງຂໍ້ມູນຂະໜາດນ້ອຍ ຈະມີໂຄງສ້າງທີ່ກະທັດຮັດກວ່າ, ເໝາະສົມກັບການຮັບ-ສົ່ງຂໍ້ມູນຂະໜາດນ້ອຍຢ່າງວ່ອງໄວ ແຕ່ຈະມີຂໍ້ຈຳກັດເມື່ອຕ້ອງຈັດການກັບການຮັບ-ສົ່ງຂໍ້ມູນຂະໜາດໃຫຍ່.

2) ປຽບທຽບປະສິດທິພາບຕາມຂະໜາດຂໍ້ມູນ:

- Pulsar ມີການເພີ່ມຂຶ້ນຂອງເວລາແບບເສັ້ນຊື່ (Linear) ເມື່ອຂະໜາດຂໍ້ມູນເພີ່ມຂຶ້ນ (ຮູບພາບທີ 2).
- RabbitMQ ມີການເພີ່ມຂຶ້ນຂອງເວລາແບບເອັກໂບເນນຊຽວ (Exponential) ເມື່ອຂະໜາດຂໍ້ມູນເພີ່ມຂຶ້ນ (ຮູບພາບທີ 2).
- ຈຸດຕັດຂອງປະສິດທິພາບຢູ່ທີ່ປະມານ 100 - 500 KB (ຮູບພາບທີ 2).

ຜົນການຄົ້ນຄວ້ານີ້ມີຄວາມສອດຄ່ອງກັບການຄົ້ນຄວ້າຂອງ Andström (2024) ທີ່ໄດ້ອະທິບາຍວ່າປະສິດທິພາບທີ່ແຕກຕ່າງກັນນີ້ມີສາເຫດມາຈາກສະຖາປັດຕະຍະກຳຜື້ນຖານທີ່ແຕກຕ່າງກັນຂອງທັງສອງລະບົບ. Pulsar ຖືກອອກແບບມາໃຫ້ມີການກະຈາຍຂໍ້ມູນແບບຂະໜານ (Parallel Distribution) ທີ່ມີປະສິດທິພາບສູງ, ເຊິ່ງເປັນປະໂຫຍດ ແລະ ເໝາະສົມສໍາລັບການຮັບ-ສົ່ງຂໍ້ມູນທີ່ມີຂະໜາດໃຫຍ່. ໃນທາງກົງກັນຂ້າມ, ການຄົ້ນຄວ້າຂອງ Sharvari & Sowmya Nag (2019) ໄດ້ຊີ້ໃຫ້ເຫັນວ່າ RabbitMQ ມີກົນໄກການຈັດການຄົວທີ່ມີປະສິດທິພາບສໍາລັບຂໍ້ມູນຂະໜາດນ້ອຍ.

3) ປະເມີນຄວາມເໝາະສົມຂອງແຕ່ລະລະບົບສໍາລັບການໃຊ້ງານໃນຕະຫຼາດຫຼັກຊັບລາວ:

- ຖ້າຂໍ້ມູນລາຄາຮຸ້ນສ່ວນຫຼາຍມີຂະໜາດນ້ອຍ (< 100 KB) RabbitMQ ແມ່ນມີຄວາມເໝາະສົມກວ່າ.
- ແຕ່ຖ້າມີການສົ່ງຂໍ້ມູນຂະໜາດໃຫຍ່ຫຼາຍ (> 500 KB) ຫຼື ຕ້ອງການຮອງຮັບການຂະຫຍາຍຕົວໃນອະນາຄົດ, Pulsar ແມ່ນມີຄວາມເໝາະສົມກວ່າ.

ໂດຍລວມແລ້ວ ຖ້າອີງໃສ່ສະພາບປະຈຸບັນທີ່ ຕະຫຼາດຫຼັກຊັບລາວ ມີພຽງ 10 ບໍລິສັດທີ່ຈິດທະບຽນຢູ່, ການນໍາໃຊ້ RabbitMQ ໃນການຮັບ-ສົ່ງຂໍ້ມູນແມ່ນມີຄວາມເໝາະສົມ. ແຕ່ໃນອະນາຄົດ ປະລິມານຂໍ້ມູນອາດມີການເພີ່ມຂຶ້ນຫຼາຍ, ດັ່ງນັ້ນ ການນໍາໃຊ້ Pulsar ແມ່ນມີຄວາມເໝາະສົມກວ່າ. ນອກຈາກນີ້, ການສຶກສາຂອງ Bellavista et al. (2014) ຍັງສະໜັບສະໜູນຜົນການຄົ້ນຄວ້າຄັ້ງນີ້, ໂດຍສະເພາະໃນແງ່ຂອງການປະຍຸກໃຊ້ໃນສະພາບແວດລ້ອມການເຮັດວຽກຕົວຈິງ. ເຂົາເຈົ້າແນະນຳວ່າການເລືອກໃຊ້ລະບົບໃດໜຶ່ງຄວນພິຈາລະນາຈາກລັກສະນະການນໍາໃຊ້ງານຕົວຈິງ ແລະ ຂະໜາດຂໍ້ມູນທີ່ຈະຮັບ-ສົ່ງເປັນຫຼັກ.

5. ສະຫຼຸບ

ການສຶກສາປຽບທຽບປະສິດທິພາບຂອງລະບົບການສົ່ງສານຂໍ້ມູນແບບ Realtime ລະຫວ່າງ Pulsar ແລະ RabbitMQ ໄດ້ສະແດງໃຫ້ເຫັນເຖິງຈຸດເດັ່ນ ແລະ ຂໍ້ຈຳກັດທີ່ແຕກຕ່າງກັນຂອງແຕ່ລະລະບົບ.

ຜົນການທົດລອງຢັ້ງຢືນວ່າ RabbitMQ ມີຄວາມເໝາະສົມກັບການນຳໃຊ້ໃນກໍລະນີທີ່ຕ້ອງການສົ່ງຂໍ້ມູນຂະໜາດນ້ອຍທີ່ຕ້ອງການຄວາມໄວສູງ, ເຊິ່ງເໝາະກັບການນຳໃຊ້ໃນລະບົບການຊື້-ຂາຍທີ່ເນັ້ນການເຮັດທຸລະກຳໄວ ແລະ ຂໍ້ມູນບໍ່ຊັບຊ້ອນ.

ໃນທາງກົງກັນຂ້າມ, Pulsar ສະແດງຄວາມສາມາດທີ່ດີກວ່າໃນການຈັດການຮັບສົ່ງຂໍ້ມູນຂະໜາດໃຫຍ່ ແລະ ສັບຊ້ອນ ຝ່າຍມີການຮັກສາປະສິດທິພາບໄດ້ຢ່າງສະໝໍ່າສະເໝີ. ຄຸນສົມບັດນີ້ເຮັດໃຫ້ Pulsar ເໝາະສົມກັບລະບົບການຊື້-ຂາຍຮຸ້ນທີ່ຕ້ອງການຄວາມໄວສູງ, ມີຄວາມໝັ້ນຄົງ ແລະ ສະຖຽນລະພາບສູງ.

ການຄົ້ນພົບນີ້ມີຄວາມສຳຄັນຕໍ່ການພັດທະນາລະບົບຊື້-ຂາຍຮຸ້ນ ໃນຕະຫຼາດຫຼັກຊັບລາວ, ໂດຍສະເພາະໃນການເລືອກໃຊ້ເຕັກໂນໂລຊີທີ່ເໝາະສົມກັບລັກສະນະການນຳໃຊ້ງານ. ຜູ້ພັດທະນາສາມາດນຳຜົນການສຶກສານີ້ໄປເປັນບ່ອນອີງໃນການພິຈາລະນາເລືອກໃຊ້ລະບົບທີ່ເໝາະສົມກັບຄວາມຕ້ອງການສະເພາະຂອງໂຄງການພັດທະນາ, ໂດຍຈະຕ້ອງຄຳນຶງເຖິງປະເພດຂອງຂໍ້ມູນ, ຂະໜາດ, ປະລິມານຂອງຂໍ້ມູນ ແລະ ຮູບແບບການນຳໃຊ້ງານເປັນຫຼັກ.

6. ຂໍ້ຂັດແຍ່ງ

ຂ້າພະເຈົ້າໃນນາມຜູ້ຄົນຄວ້າວິທະຍາສາດ ຂໍປະຕິຢາມຕົນວ່າ ຂໍ້ມູນທັງໝົດທີ່ມີໃນບົດຄວາມວິຊາການດັ່ງກ່າວນີ້ ແມ່ນບໍ່ມີຂໍ້ຂັດແຍ່ງທາງຜິດປະໂຫຍດກັບພາກສ່ວນໃດ ແລະ ບໍ່ໄດ້ເອື້ອປະໂຫຍດໃຫ້ກັບພາກສ່ວນໃດພາກສ່ວນໜຶ່ງ, ກໍລະນີມີການລະເມີດ ໃນຮູບການໃດໜຶ່ງ ຂ້າພະເຈົ້າມີຄວາມຍິນດີ ທີ່ຈະຮັບຜິດຊອບແຕ່ພຽງຜູ້ດຽວ.

7. ເອກະສານອ້າງອີງ

ຕະຫຼາດຫຼັກຊັບລາວ. (2011). *ປະຫວັດຄວາມເປັນມາຂອງຕະຫຼາດຫຼັກຊັບລາວ*. ອ້າງອີງຈາກ

<http://lsx.com.la/lo/about/history.jsp>

Andström, V. (2024). *A comparative analysis of Apache Kafka and Apache Pulsar* [Master's thesis], University of Helsinki.

Apache Pulsar. (2025). *Apache Pulsar documentation 4.0.x*. Retrieved February 1, 2025, Retrieve from <https://pulsar.apache.org/docs/4.0.x>

Bellavista, P., Corradi, A., & Reale, A. (2014). Quality of service in wide scale publish-subscribe systems. *IEEE Communications Surveys & Tutorials*, *16*(3), 1591-1616. Retrieve from <https://doi.org/10.1109/SURV.2014.031914.00192>

Chy, M. S. H., Arju, M. A. R., Tella, S. M., & Cerny, T. (2023). Comparative evaluation of Java virtual

machine-based message queue services: A study on Kafka, Artemis, Pulsar, and RocketMQ. *Electronics*, *12*(23), 4792. Retrieve from <https://doi.org/10.3390/electronics12234792>

Dobbelaere, P., & Esmaili, K. S. (2017). *Kafka versus RabbitMQ: A comparative study of two industry reference publish/subscribe implementations*. In Proceedings of the 11th ACM International Conference on Distributed and Event-Based Systems (pp. 227-238). Retrieve from <https://doi.org/10.1145/3093742.3093908>

John, V., & Liu, X. (2017). *A survey of distributed message broker queues*. *arXiv preprint arXiv:1704.00411*. Retrieve from <https://doi.org/10.48550/arXiv.1704.00411>

LATHILATH, O., KEOBOULAPHETH, S., & PHOUNNALY, P. (2020). Rate of return and risks of an investment in Lao Securities eXchange (LSX) by using CAPM. *Souphanouvong University Journal Multidisciplinary Research and Development*, *6*(3), 1382–1389. Retrieved from <https://www.su-journal.com/index.php/su/article/view/289>

Mehendale, P. (2018). Survey on real-time data processing in finance using machine learning techniques. *International Journal of Science and Research*, *8*(7), 2319-7064. Retrieved from <https://dx.doi.org/10.21275/SR24810081140>

RabbitMQ. (2025). *RabbitMQ documentation*. Retrieved February 1, 2025, Retrieve from <https://www.rabbitmq.com/docs>

Rad, B. B., Bhatti, H. J., & Ahmadi, M. (2017). An introduction to docker and analysis of its performance. *International Journal of Computer Science and Network Security*, *17*(3), 228.

Rostanski, M., Grochla, K., & Seman, A. (2014). Evaluation of highly available and fault-tolerant middleware clustered architectures using RabbitMQ. *In Federated Conference on Computer*

Science and Information Systems (pp. 879-884).
IEEE. Retrieved from
<https://doi.org/10.15439/2014F48>

Sharvari, T., & Sowmya Nag, K. (2019). *A study on modern messaging systems-Kafka, RabbitMQ and NATS streaming*. *arXiv preprint arXiv:1912.03715*. Retrieved from
<https://doi.org/10.48550/arXiv.1912.03715>

Sundar Rajan, K., Vishal, A., & Babu, C. (2021). A scalable data pipeline for real-time geofencing using Apache Pulsar. In V. Krishnamurthy, S. Jaganathan, K. Rajaram, & S. Shunmuganathan

(Eds.), *Computational Intelligence in Data Science* (pp. 3-14). Springer. Retrieved from
https://doi.org/10.1007/978-3-030-92600-7_1

Ureche, O., & Plamondon, R. (2000). Digital payment systems for Internet commerce: The state of the art. *World Wide Web*, 3(1), 1-11. Retrieved from
<https://doi.org/10.1023/A:1019217210183>.

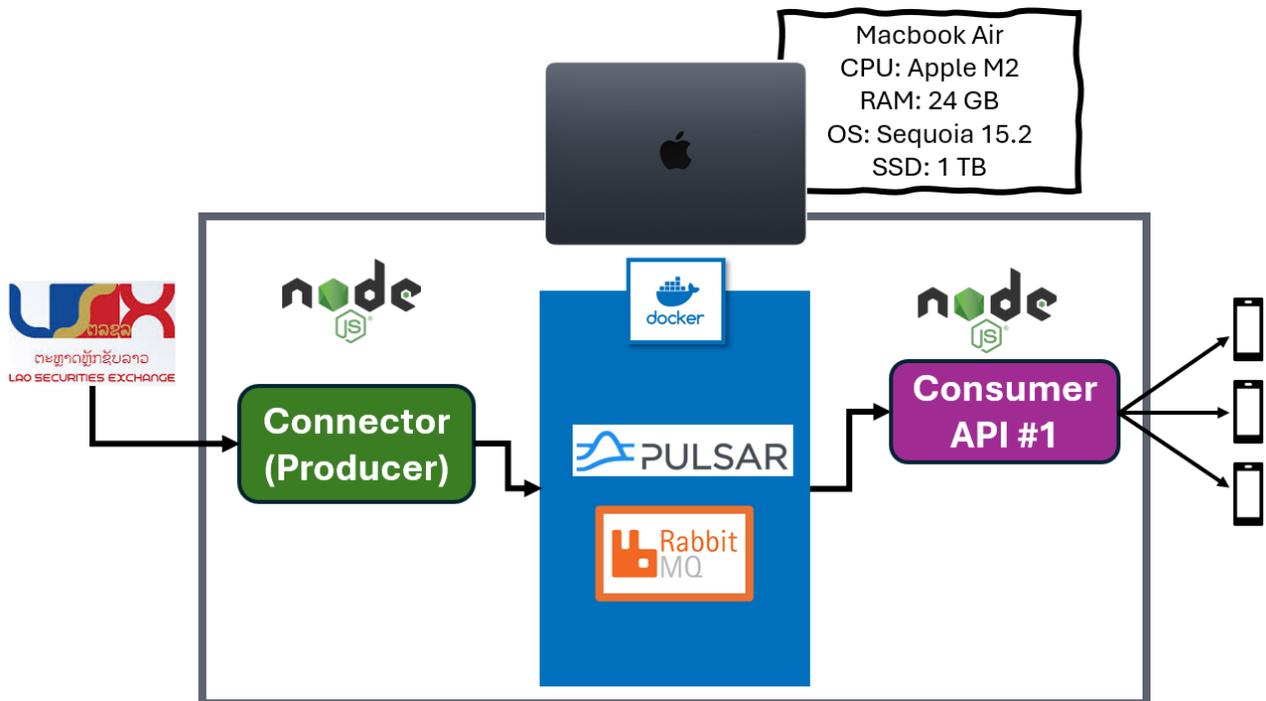
Toprak, A., Zaim, A. H., & Toprak, F. S. (2021). The effect of asynchronous message queues on the communication of IoT devices. *Veri Bilimi*, 4(3), 44-53.

ຕາຕະລາງທີ 2: ຕາຕະລາງບັນທຶກຜົນການທົດລອງຄວາມໄວສະເລ່ຍທີ່ໃຊ້ໃນການສົ່ງ-ຮັບຂໍ້ມູນຫົວໜ່ວຍເປັນ ms

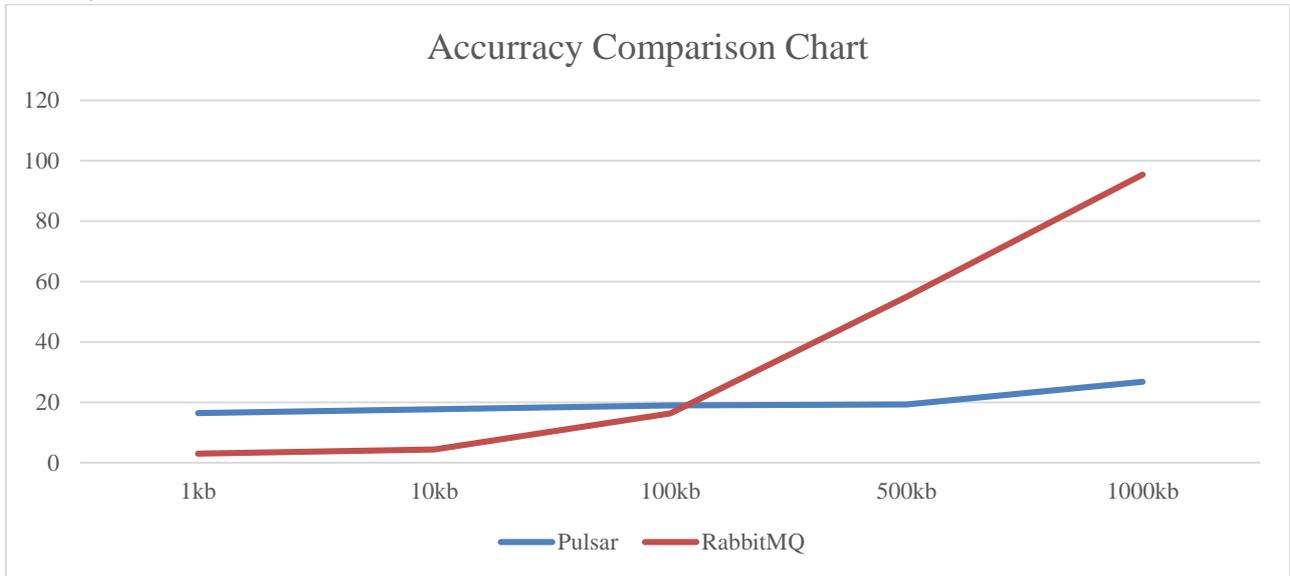
	1 KB	10 KB	100 KB	500 KB	1,000 KB
Pulsar	16.48	17.7	18.94	19.26	26.86
RabbitMQ	3.00	4.44	16.40	54.90	95.46
ຄວາມແຕກຕ່າງ	13.48	13.30	2.54	-35.60	-68.60

ໝາຍເຫດ:

- ຖ້າຄ່າຄວາມແຕກຕ່າງເປັນຄ່າ ບວກ (+) ໝາຍຄວາມວ່າ RabbitMQ ໄວກວ່າ Pulsar
- ຖ້າຄ່າຄວາມແຕກຕ່າງເປັນຄ່າ ລົບ (-) ໝາຍຄວາມວ່າ Pulsar ໄວກວ່າ RabbitMQ



ຮູບພາບທີ 1: ແຜນພາບລວມໃນການທົດລອງຮັບ-ສົ່ງຂໍ້ມູນ



ຮູບພາບທີ 2: ເສັ້ນສະແດງການສົມທຽບຄວາມໄວສະເລ່ຍໃນການຮັບ-ສົ່ງຂໍ້ມູນ ລະຫວ່າງ Pulsar ແລະ RabbitMQ