

**ANALIZA COMPARATIVĂ A UNOR SISTEME
MODERNE DE MESAJERIE**

**COMPARATIVE ANALYSIS
OF MODERN MESSAGING SYSTEMS**

**Masterand:
Vovc Artemie**

**Conducător:
prof. univ., dr. hab. Bolun Ion**

Chișinău – 2019

Ministerul Educației, Culturii și Cercetării al Republicii Moldova
Universitatea Tehnică a Moldovei
Facultatea Calculatoare, Informatică și Microelectronică
Departamentul Ingineria Software și Automatică

Admis la susținere

Șef departament: conf. univ., dr. Ciorbă Dumitru

Ciorbă Dumitru

„14” decembrie 2019

ANALIZA COMPARATIVĂ A UNOR SISTEME MODERNE DE MESAJERIE

Teză de master în
Tehnologii Informaționale

Masterand: Vovc Artemie (Artemie)
Conducător: Boben Ion (Ion)

ADNOTARE

la teza de master cu tema **Analiza comparativă a unor sisteme moderne de mesagerie** a studentului gr. TI-171M, programul „Tehnologii Informaționale”,
Artemie VOVC

Teza de master este constituită din introducere, trei capitole, concluzii, bibliografie cu șasesprezece titluri, 52 pagini de text de bază, 31 figuri și 3 tabele.

Cuvinte cheie: mesaj, producător, consumător, subiect, schimb, conexiune, mașină virtuală, limbaj de programare, scala.

Scopul tezei este de a determina cel mai performant sistem modern de mesagerie din cele Kafka, RabbitMQ, Google Cloud Platform PUB/SUB și Amazon Simple Queue Services.

Obiectivele tezei constau în: analiza sistemelor de mesagerie moderne la nivel general, analiza sistemelor la nivel arhitectural, analiza la nivel de configurări, configurarea sistemelor, configurarea mediului virtual, configurarea mediului de testare în mediul virtual, aplicarea unuia și aceluiași instrument și în aceleași condiții pentru evaluarea performanței sistemelor cercetate, realizarea la nivel de cod, testarea propriu zisă, colectarea rezultatelor, determinarea consecințelor și cauzelor și documentarea.

Obiectivele sunt realizate în baza unui scenariu. Scenariul constă în colectarea informațiilor privind durata transmiterii și primirii mesajelor în diferite formate, având o arhitectură internă similară a aplicației. Arhitectura internă prezintă un producător, un consumător și sistemul de mesagerie între aceste două componente. Arhitectura este asigurată de configurări la nivel de sistem. Conexiunile între producător, sistemul analizat și consumător sunt realizate prin una și aceeași librărie Alpakka. Aceasta asigură corectitudinea comparativă a datelor de colectat, excluzând factorul de performanță al librăriilor utilizate, deoarece se folosește una și aceeași librărie.

În baza rezultatelor calculelor, este constatat că cel mai rapid, în cazul mesajelor de mici dimensiuni, este sistemul de mesagerie RabbitMQ. Totodată, pentru mesaje de dimensiuni mari, mai performant este sistemul Kafka. Astfel, oportunitatea folosirii unuia sau a altui sistem depinde de dimensiunea mesajelor. De aceea în cazuri practice sunt necesare cercetări suplimentare ale fluxului de mesaje. De asemenea, pot fi utile cercetări comparative și a altor aspecte de performanță ale sistemelor în cauză.

ANNOTATION

to the thesis **Comparative analysis of modern messaging systems** of student from
gr. TI171M, study program „Information Technologies”,

Artemie VOVC

This thesis contains introduction, 3 chapters, conclusions, bibliography with 16 titles, 52 pages of basic text, 31 figures and 3 tables.

Keywords are: message, producer, consumer, subject, exchange, connection, virtual machine, programming language, Scala.

The thesis scope is to determine the most performed modern messaging system from: Kafka, RabbitMQ, Google Cloud Platform PUB/SUB and Amazon Simple Queue Service.

The scope of the thesis is to determine the most performed modern messaging system. The objectives of the thesis are: analysis of systems at the general level, analysis of systems at architectural level, analysis of configuration level, system configuration, configuration of virtual environment, configuration of test environment in virtual environment, application of same tool to avoid the performance factor of the used tools, implementation of test case, testing, collection of the results, determination of consequences and causes, documentation.

Objectives will be based on a scenario. The scenario is to collect the time of sending and receiving messages in different formats, having an internal architecture, which is same for each system. The internal architecture presents a producer, a consumer and messaging system between producer and consumer. Architecture is configured on system level. Connection between producer, the analyzed system and consumer is established by same library Akkka. That thing ensures correctness of data by avoiding performance factor of other third-party libraries.

Based on collected results, the best are two, for case when messages are small the best is RabbitMQ, but in other case Kafka is better. So, in most cases will be good to analyze types of messages, predict size of biggest message and make the decision based on the collected data, which system is perfect for that type of application.

CUPRINS

INTRODUCERE	8
1 ASPECTELE TEORETICE ALE SISTEMELOR DE MESAJERIE	10
1.2 SISTEMUL DE MESAJERIE KAFKA	11
1.2.1 ARHITECTURA SISTEMULUI KAFKA	11
1.2.2 SUBIECT ÎN SISTEMUL KAFKA	12
1.2.3 DISTRIBUIRE ȘI REPLICARE ÎN SISTEMUL KAFKA	14
1.2.4 GEO-REPLICARE ÎN SISTEMUL KAFKA	14
1.2.5 PRODUCĂTOR ÎN SISTEMUL KAFKA	14
1.2.6 CONSUMĂTOR ÎN SISTEMUL KAFKA	15
1.3 SISTEMUL DE MESAJERIE RABBITMQ	15
1.3.1 ARHITECTURA SISTEMULUI RABBITMQ	15
1.3.2 SCHIMB ÎN SISTEMUL RABBITMQ	16
1.3.3 TIPURI DE SCHIMBURI ÎN SISTEMUL RABBITMQ	17
1.4 SISTEMUL PUB/SUB DIN GOOGLE CLOUD PLATFORM	18
1.4.1 ARHITECTURA SISTEMULUI PUB/SUB	18
1.5 SISTEMUL AMAZON SIMPLE QUEUE SERVICE	19
1.5.1 ARHITECTURA SISTEMUL SQS	20
1.5.2 DURATA DE EXISTENȚĂ A MESAJELOR ÎN SISTEMUL SQS	20
1.6 FORMULAREA SARCINII DE CERCETARE/PROIECTARE	21
2 CONFIGURAREA MEDIULUI ȘI A SISTEMELOR DE MESAJERIE	22
2.1 CONFIGURAREA SISTEMULUI KAFKA	28
2.2 CONFIGURAREA SISTEMULUI RABBITMQ	31
2.3 MONITORIZAREA ȘI MANAGEMENTUL SISTEMELOR	32
2.4 CONFIGURAREA SISTEMULUI PUB/SUB	35

_2.5 CONFIGURAREA SISTEMULUI SQS	37
3 REZULTATELE CERCETĂRII/PROIECTĂRII	40
_3.1 COMPARAREA SISTEMELOR KAFKA ȘI RABBITMQ	40
_3.2 COMPARAREA SISTEMELOR KAFKA ȘI GOOGLE PUB/SUB	42
_3.3 COMPARAREA SISTEMELOR KAFKA ȘI AMAZON SQS	43
_3.4 TESTAREA COMPARATIVĂ	43
CONCLUZII	52
BIBLIOGRAFIE	53