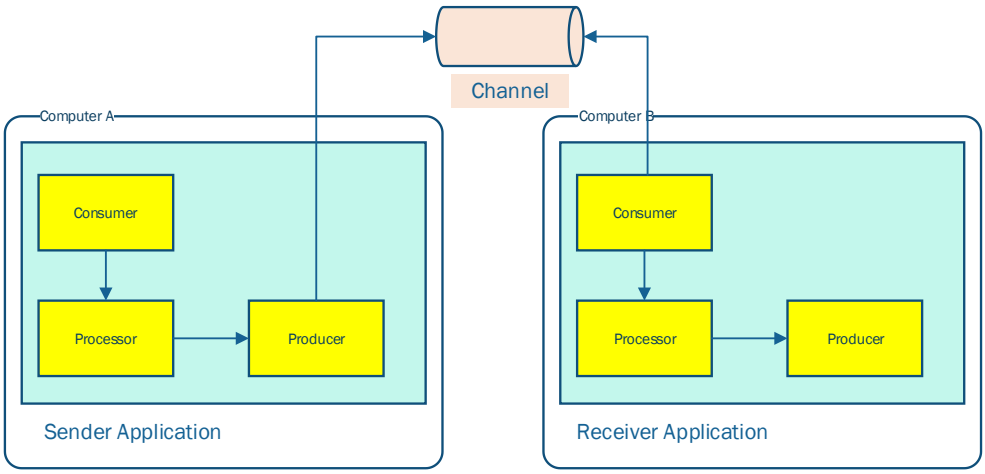
BAB I

PENDAHULUAN

1.1 Latar Belakang Penelitian

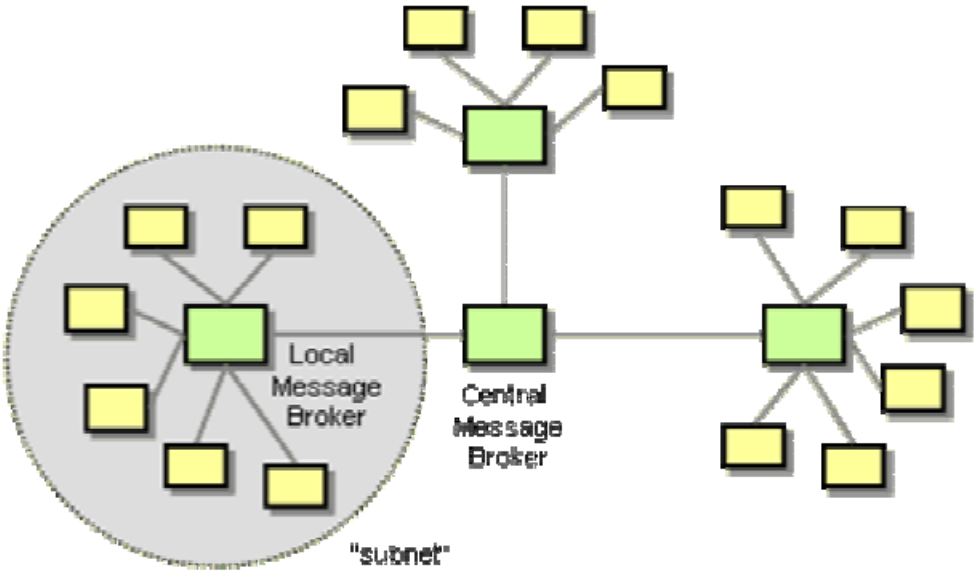
Aplikasi *enterprise* umumnya terdiri atas arsitektur *n-tier* yang terdistribusi pada beberapa komputer, menurut [8] implementasi ini kurang optimal apabila tidak dikombinasikan dengan integrasi aplikasi. Aplikasi *enterprise* terdistribusi adalah aplikasi mandiri yang sangat bergantung satu sama lain walaupun terletak pada lokasi yang berbeda. Koordinasi terpusat yang dimiliki arsitektur ini membuat arsitektur ini *tight-coupled* (tier satu bergantung pada tier lainnya). Kebalikan dari arsitektur *n-tier*, aplikasi terintegrasi berusaha meningkatkan kinerja sistem dengan membuat komponen – komponen dalam aplikasi mandiri (*standalone*) dan memiliki peran masing – masing secara khusus dan berkomunikasi secara independen, sehingga teknik pemrosesan paralel dapat dilakukan dengan lebih baik.

Salah satu contoh aplikasi *enterprise* dalam kategori sistem *messaging* adalah *middleware* berorientasi *message* (*Message Oriented Middleware*), pada dasarnya, sistem ini adalah aplikasi perantara aplikasi – aplikasi lainnya dengan fungsionalitas yang berfokus pada koordinasi dan kegiatan pengolahan serta pengiriman *message* yang memanfaatkan teknologi *messaging*. Dalam dunia sistem informasi, *middleware* berorientasi *message* menjadi perantara antar aplikasi sehingga memungkinkan adanya lingkungan yang terintegrasi satu sama lain sebagaimana yang disebutkan oleh [2] dan diperlihatkan oleh gambar di bawah ini.



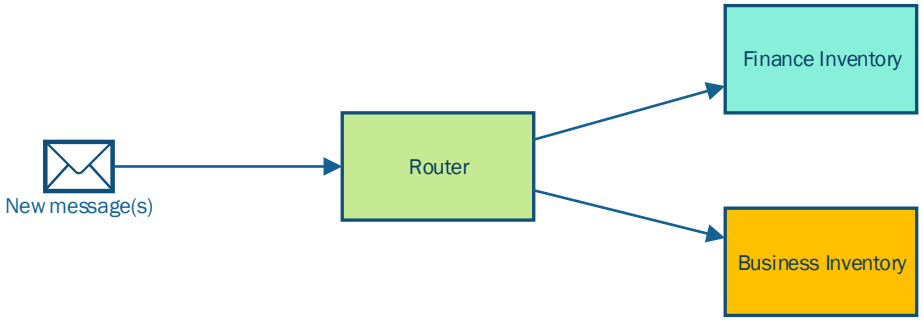
Gambar 1.1 *Middleware* berorientasi *message*

Messaging adalah istilah bagi teknologi dalam melakukan komunikasi antar sistem dengan teknik pengantaran data yang dapat diandalkan. Komunikasi antar komponen dilakukan dengan mengirimkan paket data yang disebut *message* dan memanfaatkan jalur yang disebut *channel*, bentuk lojik dari jalur ini umumnya adalah *queue* dan umumnya *sender* dan *receiver* dapat terletak pada mesin yang berbeda.



Gambar 1.2 Jaringan aplikasi *middleware* berorientasi *message* **[2]**

Esensi utama dari sebuah *middleware* berorientasi *message* adalah kemampuan *routing* atau pengiriman *message* kepada tujuan menggunakan rute yang dibentuk berdasarkan aturan – aturan tertentu.



Gambar 1.3 Fungsi *routing*

Menurut ISO 25010/2011, sebuah aplikasi *enterprise* idealnya mempertimbangkan beberapa aspek – aspek teknis dalam desain dan implementasinya. Adapun aspek – aspek tersebut adalah fungsionalitas, reliabilitas, operabilitas, kinerja, keamanan, kompatibilitas, pengaturan dan kemudahan transfer. Aspek kinerja merupakan hal yang paling sering menjadi fokus perhatian para pengembang perangkat lunak karena sifatnya yang melekat secara langsung pada pengembangan perangkat lunak itu sendiri. Kinerja sebuah perangkat lunak, dalam hal ini aplikasi *middleware* berorientasi *message*, sangat vital terlebih pada sistem yang melakukan kegiatannya secara *real time*.

Sistem *middlware* berorientasi *message* pada hakikatnya berjalan secara *asynchronous* dalam memroses *message* terhadap aplikasi – aplikasi yang terintegrasi dengannya. Desain sistem ini menekankan fokus yang kuat pada sisi *availability* dan *fault-tolerant* dari sistem *enterprise*, apabila ada kesalahan maupun kerusakan yang tidak diprediksi pada aplikasi klien, hal tersebut tidak akan berdampak (besar) pada klien yang lainnya seperti yang didisuksikan oleh [15].

Berbagai teknik dalam mencapai kinerja pemrosesan *message* dan pengantaran yang tinggi sudah banyak ditemukan dan dilakukan, di antaranya, teknik yang umum dilakukan adalah dengan menggunakan *thread*, untuk menghindari hal – hal yang berpotensial membahayakan kinerja sistem, maka digunakan *pool* untuk menggunakan ulang thread yang ada secara dinamis sehingga dapat meminimalisir *thread creation time* dan menghemat penggunaan CPU [6]. Teknik yang sama digunakan pula dalam *content-based router* dalam sistem

message broker, yaitu untuk setiap *thread* digunakan untuk pengantaran *message* ke penerima yang berbeda.

LMAX Disruptor adalah sebuah kerangka kerja teknis yang memperkenalkan cara baru dalam mengatasi keadaan yang membutuhkan *concurrent handling*. LMAX Disruptor diklaim [14] memiliki kinerja yang melampaui *multi-threading* pada *blocking queue* konvensional baik dari sisi *thoughput* maupun *latency*. Dengan kinerja yang ditawarkan di atas kertas oleh LMAX Disruptor, tentunya akan sangat mengesankan apabila kerangka kerja ini dapat dipadukan ke dalam arsitektur *content-based router* untuk dapat meningkatkan kinerjanya.

Berdasarkan artikel [12], *middleware* berorientasi *message* memiliki tantangan yang sangat krusial terutama dari segi kinerja dan skalabilitas, selain itu, variasi dari MoM itu sendiri turut menambah kompleksitas sistem. Metode dan pengukuran yang digunakan untuk mengukur kinerja MoM pada kasus dunia nyata adalah dengan mengukur *message thoughput* dan CPU *utilization*, walaupun pada kenyataannya, *memory utilization* dan IO juga kerap disertakan dalam pengukuran, namun sekali lagi, hal ini bergantung pada implementasi aplikasi MoM sesungguhnya.

1.2 Identifikasi Masalah

1.2.1 Premis

Algoritma *message routing* dengan Threadpool memiliki keunggulan mampu mengolah *task* dengan waktu yang cepat namun memiliki *overhead* pada mekanisme *locking* saat *put* dan *pull* dari *task queue* dan saat mengakses *task* dalam *queue* di *memory*

Menurut percobaan yang dilakukan [10], terdapat kemungkinan LMAX Disruptor dapat memberikan kinerja yang lebih baik dibandingkan dengan teknik pemrosesan tradisional yang menggunakan Threadpool karena tidak memiliki *overhead* pada mekanisme *locking* dan *memory contention*.

1.2.2 Hipotesa

Penggabungan teknik Threadpool dan LMAX Disruptor dapat meningkatkan kinerja *content based message router* dari sisi CPU *utilization* dan waktu pemrosesan.

1.3 Tujuan Penelitian

Tesis ini bertujuan untuk menghasilkan konfigurasi *message* router yang tepat dengan menggunakan kerangka kerja teknis LMAX Disruptor dan Threadpool.

1.4 Manfaat Penelitian

Penelitian ini diharapkan memberi perbandingan kinerja antara kerangka kerja teknis dan memberi usulan tentang perbaikan arsitektur sistem yang menggunakan teknologi serupa dan dapat diterapkan pada kasus pemrosesan *message* di hampir setiap industri yang memanfaatkan teknologi informasi untuk pengolahan data terutama yang berkaitan dengan *real time processing*, serta diharapkan dapat membantu industri tersebut untuk meningkatkan kualitas layanan menjadi lebih baik lagi.

1.5 Pembatasan Masalah dan Asumsi

Beberapa hal yang menjadi batasan masalah dalam tesis ini antara lain:

1) Implementasi pemrosesan data paralel yang dirancang dalam tesis ini berfokus pada bagian *content – based router* serta menggunakan LMAX Disruptor dan Threadpool berdasarkan kondisi tertentu.
2) Implementasi *threadpool* yang digunakan adalah milik Spring.
3) Arsitektur teknis dalam tesis ini dirancang untuk pemrosesan *message* berjumlah banyak dan berukuran relatif kecil dengan karakteristik format tertentu.
4) Analisis dilakukan dengan membandingkan kinerja sistem berdasarkan perbandingan jumlah buffer dengan jumlah *message* tertentu yang diproses di kedua kerangka teknis.
5) Bahasa pemrograman yang digunakan adalah Java.

1.6 Sistematika Penulisan

Laporan penelitian ini disusun dengan sistematika sebagai berikut:

1) Bab I Pendahuluan

   Bab ini menjelaskan tentang latar belakang penelitian, identifikasi masalah, premis dan hipotesis, tujuan penelitian, manfaat penelitian, batasan masalah dan asumsi, serta metodologi penelitian yang digunakan.

2) Bab II Tinjauan Pustaka

   Bab ini menjelaskan tentang teori dari berbagai literature yang terkait dengan topik dari penelitian ini

3) Bab III Metodologi Penelitian

   Bab ini berisi tentang analisis dari permasalahan yang ada beserta pertimbangan – pertimbangan dalam perancangan sistem yang baru.

4) Bab IV Hasil Penelitian dan Pembahasan

   Bab ini menjelaskan tentang pembahasan penelitian berdasarkan pertimbangan – pertimbangan yang dianalisis pada bab sebelumnya termasuk proses uji dan analisis hasil berdasarkan skenario yang telah dibuat.

5) Bab V Kesimpulan dan Saran

   Bab ini berisi kesimpulan dan saran dari penelitian yang telah dilakukan.