CSEN 317 – Distributed Systems

Distributed Emergency and Traffic incident Notification system

Naveena Avula - 07700009231
Sri Sai Saketh Chillapalli - 007700000089
Karumanchi Samuel - 07700009611

1. Introduction

In today's interconnected world, the rapid and accurate dissemination of emergency information and traffic incidents is crucial for public safety and efficient urban mobility. Current systems often suffer from limitations such as delayed notifications, irrelevant alerts, and lack of personalization, which can lead to confusion, inefficient resource allocation, and potentially life-threatening situations.

The proposed Distributed Emergency and Traffic Incident Notification System aims to address these challenges by leveraging cutting-edge distributed systems technology and a publisher-subscriber model. This system will serve as a centralized platform where various entities such as emergency services, traffic management centers, and authorized individuals can publish alerts and incident reports. Simultaneously, it will allow users to subscribe to receive notifications based on their geographical area, types of emergencies they're interested in, and preferred notification methods.

By employing advanced routing algorithms and real-time data processing, the platform will efficiently match and deliver relevant notifications to subscribers in real-time. This approach ensures that critical information reaches the right people at the right time, enabling swift response and informed decision-making during emergencies or traffic disruptions.

2. Problem Statement

The current landscape of emergency and traffic incident notification systems faces several challenges:

Delay in Information Dissemination: Traditional systems often suffer from delays in propagating critical information, potentially leading to increased risks during emergencies.

Lack of Personalization: Many existing systems use a one-size-fits-all approach, sending all alerts to all users regardless of relevance or personal preferences.

Scalability Issues: As user bases grow and the frequency of alerts increases, many systems face challenges in maintaining performance and reliability.

Information Overload: Users may become desensitized to alerts due to receiving too many notifications, including those that aren't relevant to them.

Lack of Integration: Often, emergency and traffic incident systems operate in silos, missing opportunities for coordinated response and comprehensive situational awareness.

Security and Trust: Ensuring the authenticity of alert sources and protecting user data are ongoing challenges in existing systems.

Our proposed Distributed Emergency and Traffic Incident Notification System aims to address these issues through innovative use of distributed systems technologies and advanced algorithms.

3. Project Objectives

The primary objectives of this project are

1. **Real-time Notification Delivery:** Develop a system capable of delivering notifications to subscribers within seconds of an incident being reported.
2. **Personalization:** Create a flexible subscription model allowing users to customize the types of alerts they receive based on their preferences and needs.
3. **Scalability and Reliability:** Design a distributed architecture capable of handling millions of subscribers

and thousands of simultaneous alerts without performance degradation.
4. **Data Security and Privacy**: Ensure the system maintains the highest standards of data security and user privacy, with robust authentication for publishers and encrypted data transmission.
5. **Fault Tolerance:** Design the system to be resilient to node failures, network partitions, and other potential issues in distributed systems.
6. **User-friendly Interfaces:** Create intuitive interfaces for both publishers (to easily create and send alerts) and subscribers (to manage their preferences and view alert history).

4. Literature Review

Our project design is informed by recent advancements in distributed systems, publish-subscribe architectures, and emergency notification technologies. Here's a review of relevant literature that will guide our implementation:

1. "**A Dynamic Failure Detector for P2P Storage System**" (IEEE, 2009) This paper discusses the challenges of failure detection in unreliable distributed systems. It proposes adaptive techniques which dynamically adjust to changing network conditions. We plan to adapt these concepts to ensure our system remains robust and can differentiate between genuine failures and temporary delays in emergency information propagation.

   Key takeaways:

   - Adaptive failure detection techniques
   - Balancing speed and accuracy in failure detection
   - Considerations for unreliable network conditions
2. "**Overlay routing network construction by introducing Super-Relay nodes**" (IEEE, 2014) This research presents an

approach to optimize routing quality in resilient and scalable networks. The concept of Super-Relay nodes could be particularly useful in our system to ensure efficient delivery of emergency notifications across wide areas.

Key takeaways:

- Use of Super-Relay nodes for improved performance
- Strategies for optimizing network topology
- Balancing between routing efficiency and network resilience

3. “**SELECT: A Distributed Publish/Subscribe Notification System for Online Social Networks**” (IEEE, 2018) This paper proposes a pub/sub notification system using a P2P network with a ring topology. While our system isn’t focused on social networks, their approach to reducing overhead by adapting connections based on user relationships could be adapted to our geographical and preference-based model.

   Key takeaways:

   - P2P network with ring topology for pub/sub systems
   - Techniques for reducing communication overhead
   - Adapting network connections based on user relationships (which we can apply to geographical proximity)

4. “**Research and design of Pub/Sub Communication Based on Subscription Aging**” (IEEE, 2018) This article introduces a subscription aging-based pub/sub communication model with prioritized subscribers and multicast transmission. These concepts could be valuable in our system, especially for prioritizing critical alerts and managing long-term subscriptions.

   Key takeaways:

   - Subscription aging mechanisms

- Prioritization of subscribers in pub/sub systems
- Efficient multicast transmission techniques

5. “**Improving the Performance of a Publish-Subscribe Message Broker**” (Rocha et al., 2019) This work explores techniques to optimize message brokers for IoT systems. Their approach to refining event handler mechanisms and optimizing communications could be directly applicable to reducing latency in our emergency notification system.

   Key takeaways:

   - Optimization techniques for pub/sub message brokers
   - Strategies for reducing communication latency
   - Scalability considerations for large-scale pub/sub systems.

6. "**EdgePub: A Dynamic Publish/Subscribe Framework for IoT at Edge**" (ACM, 2022) - This research presents an edge computing-based pub/sub system that significantly reduces latency in IoT environments. Their approach to handling mobile subscribers and dynamic topic management could be valuable for our location-based emergency notifications.

   Key takeaways:

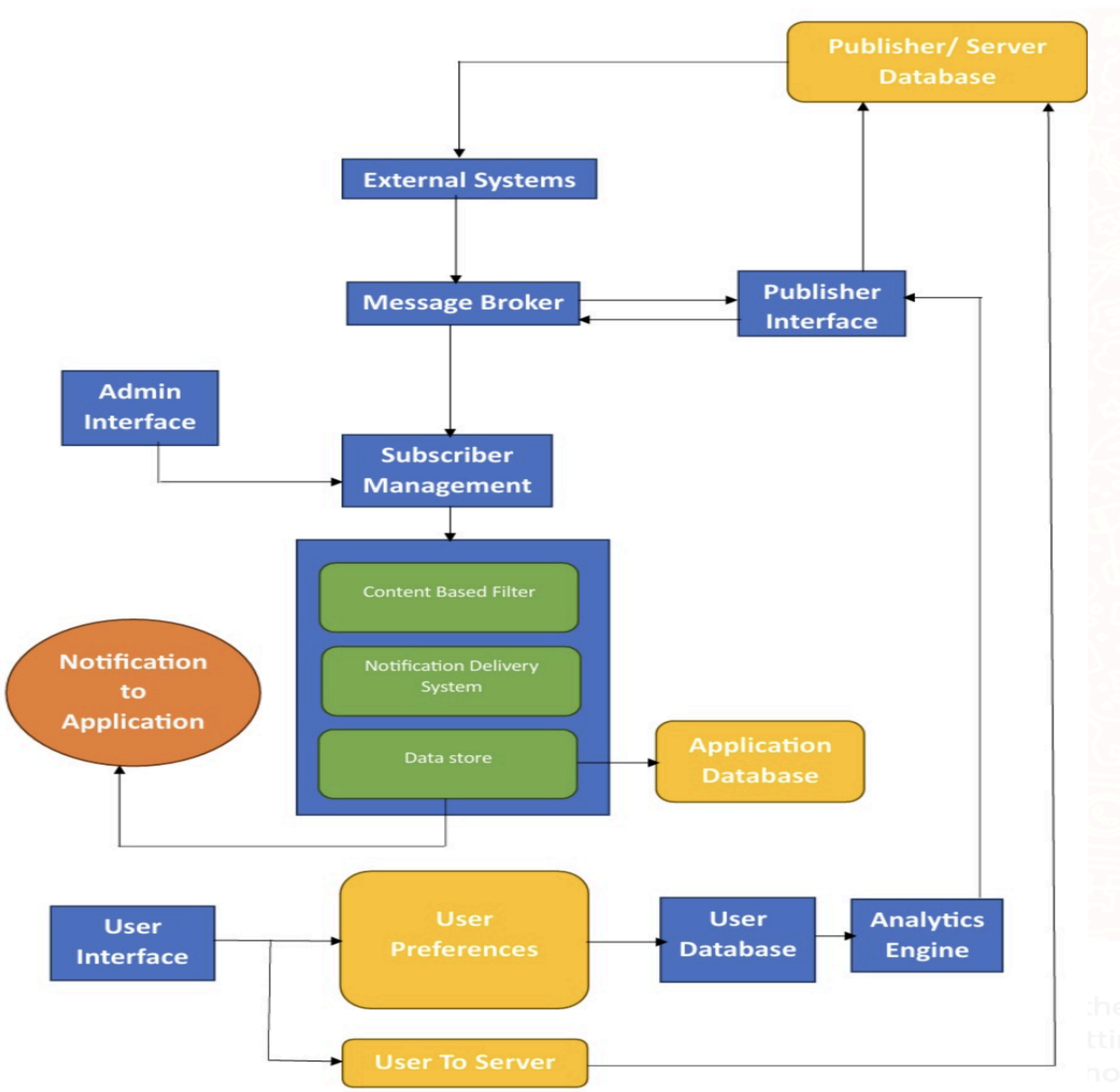
   - Edge-based architecture for reduced latency
   - Dynamic topic management
   - Mobile subscriber handling techniques

These papers provide a solid foundation for our project, offering insights into key challenges and potential solutions in areas such as failure detection, efficient routing, security, and performance optimization. We will incorporate these learnings into our system design and implementation to ensure a robust, efficient, and secure emergency notification system.

5. System Architecture

Our Distributed Emergency and Traffic Incident Notification System is designed as a scalable, fault-tolerant distributed system leveraging a publish-subscribe (pub-sub) architecture. The system is composed of several key components working in harmony to ensure rapid, accurate, and targeted delivery of critical information.

Here’s a high-level overview of the system architecture:



The main components of the system are

Publisher Interface: A secure API and web interface for authorized entities to publish alerts and incident reports.

Subscriber Management: Handles user registration, preference settings, and subscription management.

Message Broker: The core of the pub-sub system, responsible for routing messages from publishers to relevant subscribers.

Content-Based Filter: Processes notifications based on subscriber preferences and alert metadata.

Notification Delivery System: Manages the actual sending of notifications through various channels.

Data Store: A distributed database for storing user data, alert history, and system metadata.

Analytics Engine: Processes system data to provide insights and performance metrics.

Admin Interface: A web-based interface for system monitoring, management, and configuration.

User Interface: A web based application for users to interact with and subscribe to the preferred notification

The system follows a microservices architecture, with each component designed as an independent service that can be scaled and updated independently. Inter-service communication is handled through a combination of synchronous (REST APIs) and asynchronous (message queues) methods.

6. Algorithms and Technologies

Our system will leverage a variety of algorithms and technologies to ensure efficient, scalable, and reliable operation:

Content-Based Routing Algorithm

- **Purpose:** Match notification content with subscriber preferences
- Optimized for high-throughput, low-latency matching
- Support for complex Boolean queries
- Incremental updates to minimize re-computation

Failure Detection Algorithm

- **Purpose:** Detect node failures in the distributed system
- Adaptive to network conditions
- Configurable balance between speed and accuracy
- Integration with the system's self-healing mechanisms

Concurrency Control & Mutual Exclusion:

- **Purpose:** Concurrent operations can lead to race conditions and data inconsistencies. To mitigate this, we implement mutex locks, which are fundamental synchronization primitives.This algorithm provides a fair and starvation-free solution for mutual exclusion in a distributed environment.
- When a publisher wants to enter the critical section, it takes a number.
- This number is higher than all previously issued numbers.
- The publisher then announces its intent to all other publishers.
- It waits until no other publisher with a lower number (or equal number but lower ID) is trying to enter.
- It enters the critical section, performs its operation, and then exits.

Broadcast/Multicast:

- **Purpose:**For efficient dissemination of information, we employ broadcast and multicast algorithms.This algorithm ensures that all subscribers receive the message, even in the presence of failures.
- The publisher sends the message to all subscribers.
- Each subscriber, upon receiving the message, forwards it to all other subscribers.
- A subscriber delivers the message only after receiving it from a majority of subscribers.

For targeted notifications, we use a topic-based publish-subscribe system.

- Subscribers register their interests in specific topics.
- Publishers tag their messages with relevant topics.
- The system routes messages to subscribers based on topic matches.

Replication:

- **Purpose:**To ensure high availability and fault tolerance, we implement a replication strategy.This algorithm maintains multiple copies of the system state across different nodes.
- One node is designated as the primary, handling all write operations.
- Multiple backup nodes maintain replicas of the primary's state.
- The primary sends updates to all backups after each write operation.
- If the primary fails, a backup is promoted to become the new primary.
- This replication strategy ensures that the system remains operational even if individual nodes fail, providing robust fault tolerance.

By implementing these algorithms, the notification system aims to achieve high concurrency, efficient message dissemination, and robust fault tolerance.

7. Summary

The Distributed Emergency and Traffic Incident Notification System is a solution designed to address the critical need for rapid, accurate, and targeted dissemination of emergency information and traffic incidents. Leveraging a publisher-subscriber (pub-sub) model and distributed systems technologies, this platform aims to revolutionize how communities respond to emergencies and manage traffic flow.

Key features of the system include:

- Real-time, location-based notification delivery
- Scalable and fault-tolerant distributed architecture
- Customizable user preferences for alert types and severity levels

By providing a robust, scalable, and efficient platform for critical information dissemination, this system has the potential to significantly improve emergency response times, enhance traffic management, and ultimately save lives.

8.Timeline

	Task	**Target**
1	IEEE papers and Literature review	10/18/2024
2	Proposal submission	10/21/2024
3	Pub-Sub architecture design	10/28/2024
4	Development environment set up	11/04/2024

5	Deployment using Kubernetes	11/14/2024
6	Testing	11/21/2024
7	Documentation and iteration-implementing necessary changes	11/29/2024
8	Presentation	12/04/24
9	Code submission	12/04/24
10	Report	12/06/24

9.References

1. S. Jiang, X. Song, H. Wang, J. -J. Han and Q. -H. Li, "A clustering-based method for unsupervised intrusion detections," Pattern Recognition Letters, vol. 27, no. 7, pp. 802-810, May 2006

2. Shengwen Tian, J. Liao, J. Wang and Qi Qi, "Overlay routing network construction by introducing Super-Relay nodes," 2014 IEEE Symposium on Computers and Communications (ISCC), Funchal, Portugal, 2014

3. N. Apolónia, S. Antaris, S. Girdzijauskas, G. Pallis and M. Dikaiakos, "SELECT: A Distributed Publish/Subscribe Notification System for Online Social Networks," 2018 IEEE International Parallel and Distributed Processing Symposium (IPDPS), Vancouver, BC, Canada, 2018

4. D. Yang, M. Lian, Z. Zhang and M. Li, "The research and design of Pub/Sub Communication Based on Subscription Aging," 2018 IEEE International Conference on Intelligence and Safety for Robotics (ISR), Shenyang, China, 2018

5. R. Rocha, L. L. Ferreira, C. Maia, P. Souto and P. Varga, "Improving the performance of a Publish-Subscribe message broker," 2019 IEEE 22nd International Symposium on Real-Time Distributed Computing (ISORC), Valencia, Spain, 2019

6. Pham, V.-N.; Lee, G.-W.; Nguyen, V.; Huh, E.-N. Efficient Solution for Large-Scale IoT Applications with Proactive Edge-Cloud Publish/Subscribe Brokers Clustering. Sensors 2021