WhatsApp: Architectural Insights and Quality Attribute Analysis of a Global Messaging Platform

Vyshnavi Reddy Mungi
Department of Computer Science, Saint Louis University, Saint Louis, MO, USA
vyshnavireddy.mungi@slu.edu

***Abstract*— In today's world, where most of our conversations happen online, WhatsApp has become a prominent messaging app used by billions of people worldwide. It helps people communicate and exchange messages every day. With such astronomical traffic, one can't help but wonder how WhatsApp works, including its system design, software architecture, and technology. I present my analysis of how WhatsApp was able to handle so many concurrent users and messages, and the frameworks and programming languages enabled scalability while successfully leveraging its services to billions of users. This paper focuses on identifying and discussing key quality attributes and evaluates how WhatsApp's architecture effectively addresses these critical aspects.**

I. INTRODUCTION

A. History of WhatsApp

WhatsApp, established in 2009, has become one of the most influential messaging apps globally. Jan Koum and Brian Acton, the founders, came up with the idea in response to the limitations of traditional SMS messaging.WhatsApp (officially WhatsApp Messenger) is a freeware, cross-platform, centralized instant messaging (IM) and voice-over-IP (VoIP) service owned by the United States tech conglomerate Meta Platforms[1]. It allows users to send text, voice messages, and video messages, make voice and video calls, and share images, documents, user locations, and other content[1].One of the distinguishing features of WhatsApp was that it didn't rely on expensive cellular networks; it could operate over the internet. This innovation proved revolutionary, especially in regions with limited access to reliable cellular networks. Over time, WhatsApp's user base expanded exponentially, beyond the geographic boundaries and regardless of the language they speak, users adopted the application. When Facebook acquired WhatsApp in 2014, it was a major milestone, making WhatsApp even stronger in the world of communication[2]. Now, with billions of active users, WhatsApp is a major player in how people communicate online.

B. Significance of Studying WhatsApp's Architecture

WhatsApp is one of the most popular messaging apps in the world, with over 2 billion active users. It is able to handle this massive scale and volume of traffic by using a distributed architecture that is designed for scalability and reliability. Studying WhatsApp's architecture can help us to understand how to design and build other highly scalable and reliable systems.Additionally, WhatsApp is known for its strong security features, including end-to-end encryption. This means that only the sender and receiver of a message can read it, and no one else, including WhatsApp itself. Studying WhatsApp's architecture can help us to learn how to design and implement end-to-end encryption in other systems[3].Studying WhatsApp's architecture can also help us to improve the security of other messaging systems and online services.

II. ARCHITECTURAL OVERVIEW

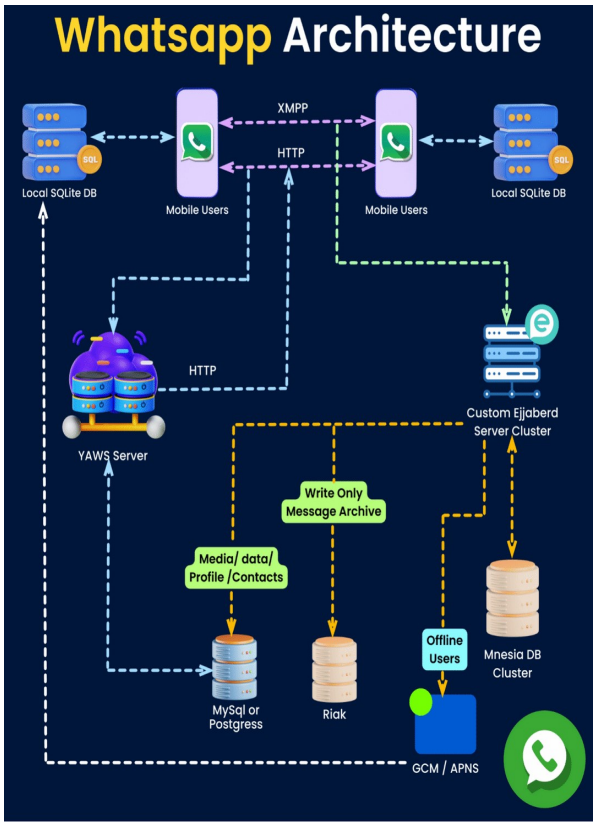

Fig. 1. Whatsapp architecture

A. Client-Server model

WhatsApp uses a client-server model, where the client is the WhatsApp app on the user's device and the server is a network of servers that handle messaging, media storage, and other tasks[4]. The client and server communicate with each

other over a secure HTTPS connection.When a user sends a message on WhatsApp, the client encrypts the message and sends it to the server. The server then decrypts the message and forwards it to the intended recipient's client. The recipient's client decrypts the message and displays it to the user.

B. Distributed Server Infrastructure

WhatsApp's server infrastructure is distributed across multiple data centers around the world. This helps to ensure that WhatsApp is reliable and scalable, even during periods of high traffic.

WhatsApp uses a variety of technologies to distribute its server infrastructure, including:

- **Load balancers:** Distribute traffic across multiple servers to improve performance and reliability.
- **Redundancy:** Replicate data across multiple servers to protect against data loss or server failure.
- **Caching:** Store frequently accessed data in memory to improve performance.
- **Content delivery networks (CDNs):** Deliver static content, such as images and videos, from servers that are close to the user's location to improve performance.

C. Database Management

WhatsApp uses a distributed database management system (DBMS) called Mnesia. Mnesia is a highly scalable and reliable DBMS that is well-suited for managing the large volume of data that WhatsApp generates.Figure 1 depicts how Mnesia DB is connected to Ejjaberd server and continues its operation. Mnesia stores data in a distributed manner across multiple servers. This helps to ensure that WhatsApp is able to handle a large number of concurrent users and transactions.In addition to storing data, Mnesia also provides a number of other features that are important for WhatsApp, such as:

- **Real-time key/value lookup:** This allows WhatsApp to quickly retrieve data from the database, even when the database is under a heavy load.
- **High fault tolerance:** Mnesia is highly fault-tolerant, that is it can continue to operate even if some of the servers in the cluster fail.
- **Dynamic reconfiguration:** Mnesia can be dynamically reconfigured to add or remove servers from the cluster. This allows WhatsApp to scale its server infrastructure up or down as needed.

WhatsApp's architecture is designed to be scalable, reliable, and secure. The distributed server infrastructure ensures that WhatsApp is available even during periods of high traffic. The database management system provides real-time key/value lookup, high fault tolerance, and dynamic reconfiguration.

Figure 1 graphically represents the architectural overview of the Whatsapp System.WhatsApp operates through a complex architecture, comprising several key components that collectively facilitate its seamless messaging experience. The Local SQLite Database serves as a temporary repository for messages on users' devices, ensuring swift access and retrieval. Meanwhile, the Mobile Users component accommodates billions of diverse users, each with their distinct interactions and preferences. The Custom Ejjaberd Server Cluster acts as the powerhouse, managing real-time communication between users. The YAWS Server ensures smooth interactions between users and servers, optimizing the flow of data. Messaging DataBases like MySQL, MySQlite, or PostgreSQL handle vast volumes of user data securely, guaranteeing robust data management. Riak serves as the backbone for storage and rapid retrieval of media and other data, contributing to the platform's efficiency. Protocols like XMPP and HTTP facilitate instantaneous messaging and data transfer. GCM/APNS protocols are responsible for pushing notifications, ensuring users stay updated across various platforms. Features like Write Only, Message Archive, and Offline Users shape the messaging experience[5]. Additionally, components like Media, Data, Profiles, and Contacts manage crucial information and media content. Lastly, HTTP serves as the bridge for web-based interactions with the WhatsApp platform, extending its functionality beyond mobile devices. Together, these components work in harmony to deliver a reliable, secure, and user-friendly messaging service to billions of users worldwide[5].

III. Key quality attributes

Software architecture quality attributes are the non-functional requirements that define how well a software system should perform. By definition, a quality attribute (QA) is a measurable or testable property of a system that is used to indicate how well the system satisfies the needs of its stakeholders beyond the basic function of the system[6]. WhatsApp has the following key qualities: portable, reliable, secure, scalable, performant, available, deployable, modifiable, and usable. These qualities ensure that the system performs well beyond its basic functionality.

The key quality attributes and their importance in the WhatsApp software system are:

- **Portability** is the ability of a software system to be moved from one hardware or software platform to another without significant changes. This is important for systems that need to be deployed on a variety of different environments, such as mobile devices, cloud platforms, or on-premises servers.WhatsApp is available on a wide range of platforms, including Android, iOS, Windows, macOS, and KaiOS, since WhatsApp is primarily developed in Java, which is a portable language that allows applications to run on a wide range of devices and operating systems[3].
- **Reliability** is the ability of a software system to perform its intended function correctly and consistently over a period of time. This is important for systems that need to be available and responsive to users, even under heavy load or in the event of errors[6]. WhatsApp is highly reliable, even on slow or unreliable internet connections. This is because it uses a variety of techniques to reduce bandwidth usage and improve resilience to errors.

- **Scalability** is the ability of a software system to handle increasing amounts of data or traffic without sacrificing performance or reliability. This is important for systems that need to be able to grow with their user base or data requirements. WhatsApp is designed to scale to handle billions of users and messages. It uses a distributed architecture to distribute the load across multiple servers. This helps to ensure that WhatsApp remains performant and reliable even as the number of users and messages increases.
- **Security** is the ability of a software system to protect itself from unauthorized access, use, disclosure, disruption, modification, or destruction of information. This is important for systems that handle sensitive data, such as financial or medical information[6]. WhatsApp is one of the most secure messaging apps available. It uses end-to-end encryption to protect messages from being read by anyone other than the sender and recipient. This means that even if WhatsApp is hacked, the attacker will not be able to read any of the messages that have been sent or received.
- **Performance** is the speed and efficiency with which a software system can respond to user requests and complete its tasks. This is important for systems that need to be able to provide a good user experience and handle high volumes of traffic. WhatsApp is very performant, even on low-end devices. This is because it is carefully optimized for speed and memory efficiency. For example, WhatsApp uses a technique called "lazy loading" to only load images and other media when they are needed by the user. This helps to reduce memory usage and improve performance.
- **Availability** is the amount of time that a software system is up and running and accessible to users[6]. This is important for systems that need to be available 24/7 or for systems that are critical to business operations. WhatsApp is highly available, with a global uptime of over 99%. This is because it is hosted on a redundant infrastructure. This means that if one server goes down, the traffic is automatically routed to another server. This helps to ensure that WhatsApp is always available to users.
- **Deployability** is the ease with which a software system can be deployed into production[6]. This includes tasks such as building, testing, and packaging the system, as well as configuring and running the system on the target environment. WhatsApp is easy to deploy on new servers. This is because it is packaged as a containerized application. This means that WhatsApp can be deployed with minimal effort on any server that supports Docker.
- **Modifiability** is the ease with which a software system can be changed or extended to add new features or fix defects. This is important for systems that need to be adaptable to changing requirements or for systems that need to be maintained and updated over time. WhatsApp is easy to modify and extend with new features. This is because it is designed with a modular architecture. This means that new features can be added or removed without affecting the rest of the system.
- **Usability** is the ease with which users can learn and use a software system. This is important for systems that need to be accessible to a wide range of users, including those with limited technical expertise. WhatsApp is very easy to use, even for people with limited technical expertise. This is because it has a simple and intuitive user interface. For example, WhatsApp uses a simple chat interface that is familiar to most users.

IV. Software Architecture Addressing the Identified Quality Attributes

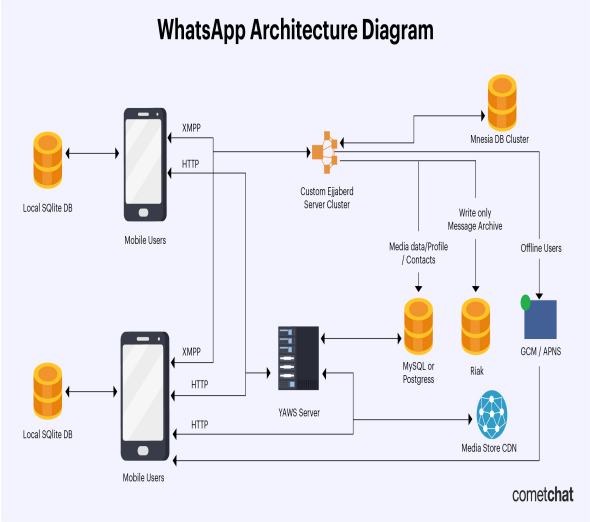

Fig. 2. Whatsapp architecture in Component and Connector structural view

A. Most prominent quality attributes

*1) **Portability**:* WhatsApp's portability allows it to run on a wide range of platforms, including iOS, Android, desktop, web, and Windows Phone. This is achieved by building native apps for each platform, meaning that each app is written in the programming language that is native to the platform it is running on. This makes it easier to develop and maintain WhatsApp, and it also ensures that the app runs well on each platform. Here's a list of all the supported platforms with the front-end language(s) that were used to build each one[3]:

- Android: Java
- iOS: Swift
- Windows Phone: C#
- Web app: JavaScript/HTML/CSS
- Mac Desktop app: Swift/Objective-C
- PC Desktop app: C/C#/Java

*2) **Reliablity and Scalability**:* WhatsApp ensures reliability by using Erlang, a programming language designed to be concurrent, scalable, and reliable, and uses YAWS, a web server designed to be reliable and scalable. Erlang's process-based model allows for fault isolation, self-healing, and horizontal scaling, while YAWS uses load balancing,

distributed databases, and WebSockets to handle high traffic and maintain uptime[3],[4].

*3) **Security**:* The security quality attribute in WhatsApp's encryption technology is primarily characterized by its implementation of end-to-end encryption. This means that the message is encrypted by the sender and can only be decrypted by the intended recipient, ensuring that no intermediary, including WhatsApp itself, has access to the message content in its readable form. WhatsApp employs the Signal Encryption Protocol, a robust and open-source encryption protocol designed for secure messaging[8]. This protocol guarantees that even if messages are temporarily stored on WhatsApp servers for delivery purposes, they remain encrypted and unreadable to anyone without the appropriate decryption keys. While end-to-end encryption provides a strong foundation for user privacy, it's essential to acknowledge that the effectiveness of this security measure can still be influenced by other factors, such as user behavior and device security.

*4) **Performance**:* By storing chat locally, WhatsApp is able to avoid downloading all the messages from the cloud every time we open the app. This saves resources and improves performance.However, this also means that WhatsApp is less independent of the platform it is running on and more susceptible to resource contention. If the device is running low on resources, WhatsApp may not be able to access the chat database as quickly, which could lead to performance problems.Overall, the decision to store chat locally is a trade-off between performance and independence. WhatsApp has chosen to prioritize performance, as this is more important to most users. WhatsApp uses a highly modified version of XMPP(Extensible Messaging and Presence Protocol) on an Ejabberd server as a messaging protocol and HTTP(Hypertext Transfer Protocol) as a communication protocol to communicate with clients. Devices connect using both the HTTP and XMPP protocols. HTTP is commonly used for web services, while XMPP is tailored for real-time chat applications. The XMPP on the client opens an SSL(Secure Sockets Layer) socket to the WhatsApp servers. All the sent messages are queued on the servers until the client opens or reconnects to this socket to retrieve the messages. WhatsApp's use of a highly modified version of XMPP on an Ejabberd server can be seen as a trade-off between performance and security. On the one hand, using XMPP allows WhatsApp to take advantage of its decentralized nature. This means that WhatsApp servers do not have exclusive control over the data, which can improve performance and reliability. In addition to this, WhatsApp chose FreeBSD as its operating system because of its high performance, especially when it comes to system load per packet[3],[7]. This is important for WhatsApp because it needs to be able to handle a high volume of traffic with minimal latency.

*5) **Availability**:* WhatsApp uses Ejabberd, Mnesia, and BEAM to achieve high availability and fault tolerance. Ejabberd is a highly available XMPP server. Ejabberd monitors the heartbeats of its components. If a component does not respond to a heartbeat, Ejabberd knows that it has failed, along with this it logs all occurrences in its component. This information can be used to identify and troubleshoot faults. Ejabberd has redundant infrastructure in place. This means that if one server or component fails, there is another one to take its place. Mnesia is a distributed database designed for high availability, Mnesia replicates data across multiple servers. If a server fails, Mnesia can continue to operate using the data on the other servers, i.e Mnesia supports dynamic reconfiguration[3]. BEAM(Bogdan's Erlang Abstract Machine) is a virtual machine designed for highly concurrent and fault-tolerant applications.BEAM is so crucial to the WhatsApp system design that the WhatsApp team has published many patches and fixes to the core source code.WhatsApp also uses other techniques such as load balancing to improve its availability.

*6) **Deployability**:* Deployability is important for WhatsApp software because it allows businesses and organizations to use WhatsApp in a way that best suits their needs.WhatsApp software can be designed to support a variety of platforms and devices, including on-premises servers, cloud servers, and mobile devices. This allows businesses and organizations to deploy the software on the platforms and devices that they are already using. WhatsApp provides a number of deployment tools and documentation to help businesses and organizations deploy their WhatsApp software. In addition to providing this versatile deployment platforms, WhatsApp uses microservice architecture to implement its messaging service, backend infrastructure, and other services, which makes the deployment easier [6],[9],[10]. For example, WhatsApp's messaging service is implemented as a microservice that communicates with other microservices, such as the user authentication service and the message delivery service. This allows WhatsApp to scale its messaging service independently of its other services, and to deploy new features to its messaging service without affecting its other services.

*7) **Modifiability**:* WhatsApp software is designed to be modifiable in a number of ways. The software is modular, meaning that it is made up of separate components that can be easily swapped out or replaced. This makes it easy to add new features or fix bugs without having to rewrite the entire system. For instance, in Whatsapp Mnesia is the only DBMS that's written in Erlang. This in itself is a benefit because there are no data structure differences between Erlang in the application and Erlang in the DBMS. Coding is, therefore, quicker and more explicit, and hence the modifiability makes the task easier[9].In addition, WhatsApp software is well-documented, meaning that there is clear and concise documentation available for all of the system's components. This makes it easy for developers to understand how the system works and to make changes to it without breaking anything.

*8) **Usability**:* WhatsApp prioritizes usability by offering a user-friendly interface, making messaging effortless. The platform seamlessly integrates multimedia sharing, contact management, and status updates, ensuring a user-friendly

experience. Additionally, robust security and privacy settings are easily accessible, providing users with control over their information[11]. WhatsApp's commitment to regular updates further enhances usability, making it a widely preferred choice for effortless and efficient communication across the globe.

B. Least prominent quality attributes

While the least prominent quality attributes include: safety, integrability, testability.

1) Safety:

- Less Critical for Messaging Apps: Safety, which includes aspects like fault tolerance and error handling, may not be as critical for a messaging app like WhatsApp compared to applications in high-risk domains (e.g., aviation, healthcare). The consequences of failures in a messaging app are generally less severe, making safety a less prominent concern.

2) Integrability:

- Stand-Alone Application: WhatsApp is primarily a stand-alone application designed for mobile devices. It is not intended to be deeply integrated with other systems or applications. While it does allow integration with third-party services through APIs, this is not a core feature for most users.

3) Testability:

- Well-Established Functionality: Messaging apps like WhatsApp typically have well-defined and stable functionalities. Extensive testing is still crucial, but the level of complexity and risk associated with untested features is lower compared to applications that control critical systems or handle sensitive information.

V. CONCLUSIONS

WhatsApp's software architecture excels in meeting a range of crucial quality attributes. It ensures portability across various devices, offering a reliable and secure messaging experience. The platform's robust architecture allows for seamless scalability and high performance, accommodating its extensive user base. Availability is maintained through redundant servers, while deployability is facilitated by straightforward installation processes. The platform's modifiable architecture supports easy updates and feature additions. Lastly, WhatsApp's user-friendly interface and distinct features prioritize usability, contributing to its global popularity. Overall, the software architecture effectively addresses these key quality attributes, underscoring WhatsApp's position as a leading messaging platform.

.

REFERENCES

[1] Wikipedia Contributors, "WhatsApp," Wikipedia, Apr. 13, 2019.

[2] S. N. and E. Sherly, "WhatsApp Calling: a Revised Analysis on WhatsApp's Architecture and Calling Service," International Journal of Engineering and Computer Science, vol. 5, no. 6, pp. 17013-17016, June 2016.

[3] Cosette Cressler, "WhatsApp's Architecture and System Design," CometChat Blog, Nov. 2, 2021. [Online]. Available: https://www.cometchat.com/blog/whatsapps-architecture-and-system-design.

[4] admin, "WhatsApp Architecture: Software Development Architecture - TechDotBit," TechDotBit, Oct. 03, 2023. http://techdotbit.com/blogs/tech/whatsapp-architecture-software-development-architecture/ (accessed Oct. 28, 2023).

[5] "Santosh Kumar Mishra on LinkedIn," www.linkedin.com. https://www.linkedin.com/posts/iamsantoshmishra_systemdesign-whatsappdesign-interview-activity-7103942182167797761–0P/ (accessed Oct. 28, 2023).

[6] L. Bass, P. Clements, and R. Kazman, Software Architecture in Practice, 4th Edition. 2021.

[7] Quastor, "How WhatsApp scaled to 1 billion users with only 50 engineers," Quastor. https://blog.quastor.org/p/whatsapp-scaled-1-billion-users-50-engineers.

[8] Wikipedia Contributors, "Signal Protocol," Wikipedia, Oct. 16, 2019. https://en.wikipedia.org/wiki/Signal_Protocol.

[9] "System Design: WhatsApp," DEV Community. https://dev.to/karanpratapsingh/system-design-whatsapp-fld.

[10] "Monoliths and Microservices | System Design," Karan Pratap Singh, Oct. 05, 2022. https://www.karanpratapsingh.com/courses/system-design/monoliths-microservices (accessed Oct. 28, 2023).

[11] D. F. Dhahir, "The usability of Whatsapp Messenger as online teaching-learning media," Journal of Information Technology and Its Utilization, vol. 3, no. 2, p. 48, Dec. 2020, doi: https://doi.org/10.30818/jitu.3.2.3629.

[12] Ç. Güler, "Use of WhatsApp in Higher Education," Journal of Educational Computing Research, vol. 55, no. 2, pp. 272–289, Sep. 2016, doi: https://doi.org/10.1177/0735633116667359.