WhatsApp is a **widely used messaging application** that enables users to send texts, make voice and video calls, and share various media files. Its design focuses on **speed, reliability, and security** to ensure a seamless communication experience. Let's break down its **architecture** and the **technologies** it employs in simple terms.

**1. Client-Server Architecture:**

WhatsApp operates on a **client-server model**. In this setup, your **device (client)** communicates with **WhatsApp's central servers** to send and receive messages. When you send a message, it's first transmitted to the server, which then forwards it to the recipient's device. This method ensures efficient and reliable message delivery.

**2. Programming Languages and Technologies:**

* **Erlang:** WhatsApp's backend is primarily built using **Erlang**, a programming language known for handling numerous simultaneous connections efficiently. This makes it ideal for real-time applications like WhatsApp.
* **Extensible Messaging and Presence Protocol (XMPP):** WhatsApp utilizes a customized version of **XMPP**, an open standard communication protocol designed for instant messaging. This protocol facilitates the exchange of messages and presence information between clients through the server.
* **Java/Swift:** Java for Android apps, Swift for iPhones.

**3. Data Storage:**

* **SQLite:** On your device, WhatsApp uses **SQLite**, a lightweight database, to store messages and media locally. This allows you to access your chat history even when offline.
* **Mnesia:** On server side, Stores user data (like contacts) and works super fast with Erlang.
* **Server Storage:** Messages are temporarily stored on WhatsApp's servers using a "store and forward" mechanism. Once the message is delivered to the recipient, it's deleted from the server. If undelivered after 30 days, it's also deleted.
* **Temporary Storage:** If a message cannot be delivered immediately (for example, if the recipient is offline), it stays on the server for a short time and is then sent when possible.

**4. Security Measures:**

* **End-to-End Encryption:** WhatsApp implements **end-to-end encryption**, ensuring that only the sender and recipient can read the messages. Not even WhatsApp can access the content of the messages, enhancing user privacy.
* **Signal Protocol:** WhatsApp uses the **Signal Protocol** for encryption, ensuring that your messages remain **private** and **secure**.

**5. Multimedia Handling:**

* **Media Upload and Retrieval:** When you send media files like photos or videos, they're uploaded to an HTTP server, and a link to the content is sent to the recipient along with a thumbnail. This approach optimizes the delivery of large files.

**6. Voice and Video Calls:**

* For calls, WhatsApp uses technologies like WebRTC (for real-time communication) and the Opus Codec (for high-quality audio), ensuring clear and smooth calls.

In summary, WhatsApp's architecture is designed to provide a **fast, reliable, and secure** messaging experience. By leveraging technologies like **Erlang**, **XMPP**, and **end-to-end encryption**, it ensures efficient communication while maintaining user privacy.