1 Google Cloud Pub/Sub Components

Pub/Sub[3] is an asynchronous and scalable messaging service provided by Google Cloud. It is commonly used for streaming analytics and data integration pipelines to load and distribute data, and it integrates smoothly with other Google Cloud services.

In our project, Pub/Sub acts as a bridge between the Cloud Run component and the local computer. Cloud Run handles heavy-duty data processing, while the local computer performs further processing on more fine-grained and sensitive data. This combination is a common approach in modern big data architecture[5].

Pub/Sub belongs to a group of messaging systems called messaging-oriented middleware[3], such as KafkaMQ[2] and RabbitMQ. These systems share features such as asynchronous communication, persistent storage, fault tolerance, and scalability through clustering and horizontal expansion.

Another common communication mechanism in distributed systems is Remote Procedure Call (RPC)[7], which allows functions on remote machines to be called as if they were local. RPC is suitable for immediate, fine-grained operations like cloud APIs and control-plane tasks. However, it is less suitable for big data processing, because all data must be fully packaged, sent, and processed as a single unit[1], which becomes inefficient as data volume grows.

Messaging-oriented middleware, on the other hand, is designed for high-throughput pipelines. It avoids the bottlenecks of RPC by using asynchronous communication, lightweight serialization, and log-based storage, allowing large volumes of data to flow efficiently without per-call overhead[6].

What makes it even more suitable for our project is that Pub/Sub implements the publish–subscribe pattern, where consumers subscribe to specific messages from producers. This pattern reduces dependencies between processes and enables highly decoupled systems[7].

In financial trading scenario, high-volume real-time data is passed through system, and it has to be integrated across heterogeneous sources. The Pub/Sub pattern simplifies this integration, allowing systems in different languages or technologies to communicate without tight coupling[4].

Overall, Google Cloud Pub/Sub provides an asynchronous, scalable, and fault-tolerant foundation for high-volume financial data processing. By decoupling publishers and subscribers, it overcomes the limitations of RPC for large datasets and efficiently connects Cloud Run and local processing components. In financial trading systems, Pub/Sub enables real-time, flexible, and reliable integration across heterogeneous subsystems.

References

[1] Andrew D. Birrell and Bruce J. Nelson. Implementing remote procedure calls. *ACM Transactions on Computer Systems*, 2(1):39–59, 1984.

[2] The Apache Software Foundation. *Apache Kafka — Documentation*, 2025. "Commit Log / Overview" section, accessed Nov 29 2025.

[3] Google. What is pub/sub? https://docs.cloud.google.com/pubsub/docs/overview, 2025. Last updated 2025-11-24 UTC.

[4] Donald MacKenzie. Middleware's message: The financial technics of codata. *Philosophy & Technology*, 32(1):123–145, 2019.

[5] Dominik Ryzko. *Modern Big Data Architectures: A Multi-Agent Systems Perspective*. Wiley, Hoboken, NJ, 2019.

[6] Team IT Security. How kafka achieves high throughput: A breakdown of its log-centric architecture, 2024. Accessed: 2025-03-13.

[7] M. van Steen and A.S. Tanenbaum. *Distributed Systems, 4th ed.* 2023.