PERFORMANCE BENCHMARK

—

Proof of 1 million transactions per second (TPS) in the public cloud

Table of Contents

Executive Summary

Totogi demonstrates an industry-leading, never-seen-before, one million TPS 5G network traffic benchmark on the public cloud.

Totogi is the only true webscale public cloud charging solution that utilizes public cloud technologies and innovations to provide a charging framework that scales to billions of subscribers/devices and millions of transactions per second.

This benchmark summarizes the platform while linearly scaling across subscribers and transactions in a single multi-tenant data schema. The Totogi solution scaled subscribers from 5 million to 500 million, and transactions were scaled from 5,000 to 500,000 per second, with no impacts. This use case also demonstrated a multi-region peak load of 1 million transactions per second to show how the platform scales globally without hitting any physical limits, unlike traditional on-premise, single data center systems.

In addition, it is priced at a cost per transaction of $0.00005 ($50 per million transactions), which is 80% lower than traditional on-premise systems. This is all done while providing an accurate idempotent ACID-compliant processing model that includes synchronously replicating transactions across three data centers for the ultimate high availability solution.

Webscale Charging from Totogi

Totogi has redefined how charging should be done using public cloud technologies from Amazon Web Services (AWS). There are three primary cloud technologies that Totogi is built on - Graviton (ARM) processors, DynamoDB NoSQL database, and ubiquitous global data centers connected via high-speed networks.

One of the key elements in providing high performance at a low cost is the Graviton Arm processor. It is both faster and cheaper than the traditional Intel/AMD processor. For example, see this third party benchmark comparing **Graviton2 against AMD and Intel.**

Another critical component is Amazon DynamoDB. All other charging engines use a SQL or in-memory database, which is why charging engines have struggled for decades with achieving the scale needed for CSPs. By using DynamoDB, Totogi breaks that traditional bottleneck. It requires both the NoSQL paradigm combined with the ubiquitous global data centers to deliver this unmatched performance and scale.

As per Amazon:

"DynamoDB is a fully managed NoSQL database service that provides fast and predictable performance with seamless scalability. DynamoDB automatically spreads the data and traffic over a sufficient number of servers to handle your throughput and storage requirements while maintaining consistent and fast performance. All of your data is stored on solid-state disks (SSDs) and is automatically replicated across multiple Availability Zones in an AWS Region, providing built-in high availability and data durability."

Totogi selected DynamoDB as the leading NoSQL database in the public cloud. The characteristics that were an important part of the evaluation were as follows:

Traditional SQL	DynamoDB
Optimized for storage	Optimized for compute
Normalized/relational	Denormalized/hierarchical
Ad hoc queries	Instantiated views for known patterns
Scale vertically	Scale horizontally
Good for OLAP	Built for OLTP at scale
ACID transactions	ACID transactions
Optional asynchronous replication	Built-in synchronous replication across at least 3 data centers

Database transactions are written directly to tables by applications; however, under the hood, the tables are sharded to scale as needed horizontally. This allows Totogi to provide a single, multi-tenanted platform that can manage every subscriber from every telco in the world. The majority of the existing "high-performance" charging solutions in the market today hit hard limits in terms of database size. This directly impacts business logic, such as the number of subscribers who can share balances.

A single DynamoDB table is replicated across multiple data centers within a region. Each data center is entirely independent with its redundant power and networking. Within a region, data centers are located kilometers from each other but interconnected with dedicated low latency networks. When data is written to DynamoDB, it is synchronously replicated to a majority of data centers. Even if multiple data centers simultaneously become unavailable or destroyed in a natural disaster, DynamoDB remains fully available without a single lost transaction. Taking advantage of DynamoDB's replication allows other portions of Totogi's charging to be spread across these same data centers without performance penalties, resulting in a native three-way active deployment. In addition, DynamoDB supports asynchronous replication between regions, allowing Totogi to maintain a single view of data throughout the world and provide speedy disaster recovery in the event an entire region is destroyed or unavailable.
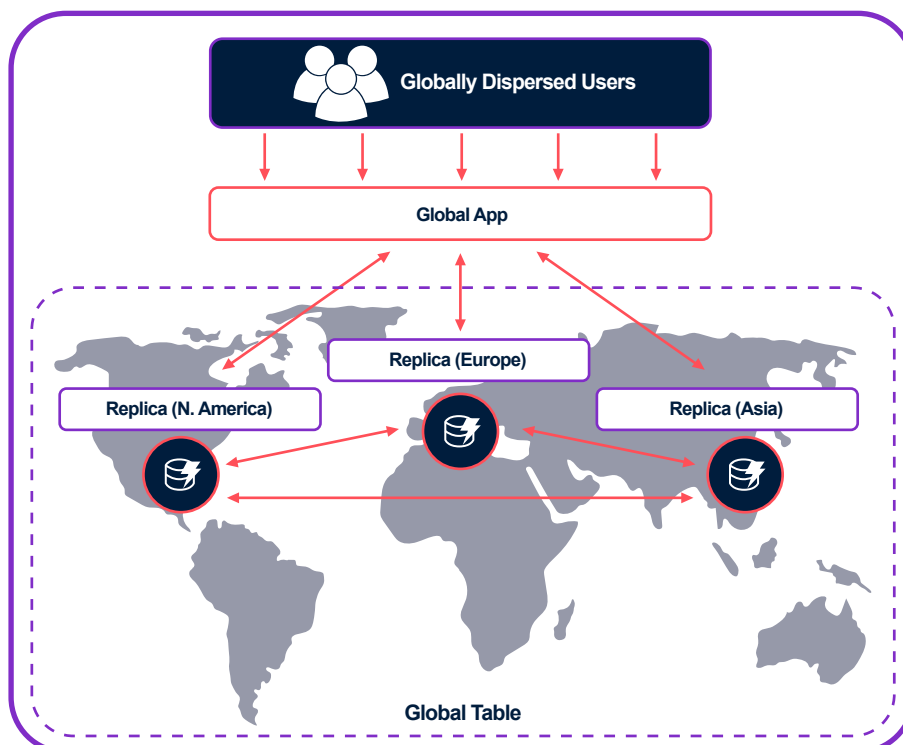
Planetary Scalability and Commodity Pricing

Totogi is designed to scale globally. Totogi customers can benefit from a free tier for up to 500M transactions per month, and pay only $0.00005 per transaction for usage beyond this level.

Scalability and Performance Test Results

In a recent test conducted on AWS with a provisioned test subscriber base of 500 million mobile subscribers, the Totogi charging platform demonstrated true webscale performance at a breakthrough price point.

Test Setup

To test traffic against 500M subscribers, a test driver generated charging requests using the Gatling framework to orchestrate the traffic. The Totogi engine comprised a cluster of c6g.8xlarge AWS EC2 instances. The underlying database consisted of a single DynamoDB table with a separate table used for CDR storage.

Methodology

Mixed charging traffic was generated by a traffic generator and sent to the Totogi charging platform using the industry standard 5G CHF interface (N40). To truly demonstrate webscale performance, 500M subscriber accounts were provisioned in a single DynamoDB instance that was replicated across two Amazon regions for high availability. Within each region, three active data centers were deployed to process incoming transactions.

Configured Price Plans

Four price plans were configured to represent real-world scenarios of prepaid, postpaid, and sharing.

Traffic Profile

Traffic Type	Description
Voice 1	Create and Release operations (60 seconds duration)
Voice 2	Create, Update and Release operations (180 seconds duration, 90 seconds between operations)
Data	Create, Update and Release operations (24 minutes duration, 180 seconds between operations)
Messaging	Create operation only

Software Environment

The application software tested consisted of:
- Totogi v1

Deployed on:
- Amazon Linux 2; based on the Linux 4.14 kernel

Hardware Environment and Deployment Architecture

The deployment architecture is shown below. A dual region, trans-Atlantic approach, replicates real-world scenarios and demonstrates extreme availability not seen before by other charging vendors.
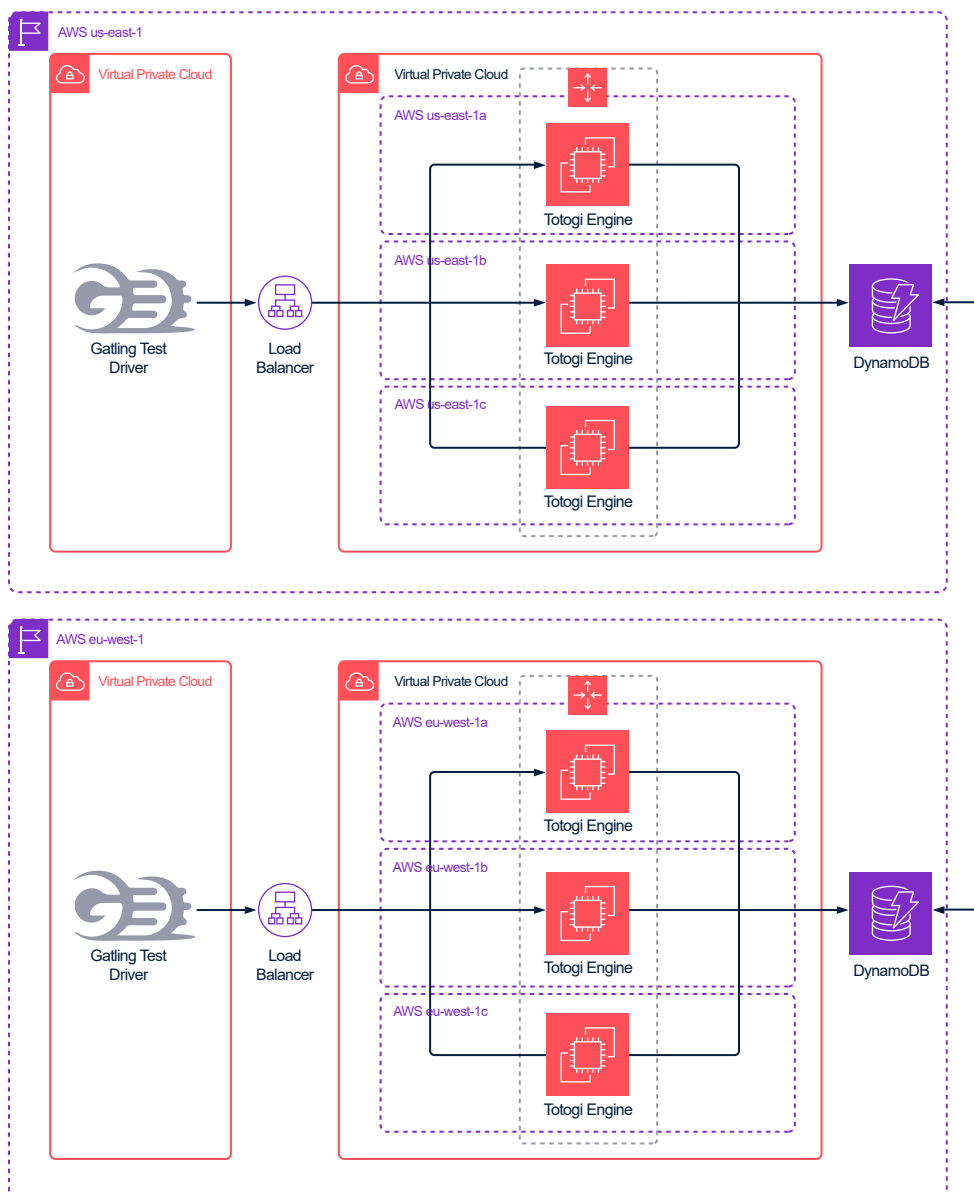
Test Results

Totogi demonstrated consistent performance with linear scalability across all tests. A low error rate of 1 error per 3 million transactions was observed and as per AWS guidance, retry logic ensured transactions were subsequently processed successfully.

Efficient Resource Utilization

CPU utilization was kept at around 30% of capacity to support a highly available and redundant architecture. While this requires more resources in terms of CPU cores, it has the added advantage that there is no impact if an EC2 instance or an entire availability zone goes down.

Linear Scalability

The benchmark ranged from 5M subscribers and 5K TPS up to a dual region 500M subscribers, 1M TPS, shown in the table below.

	5M	50M	500M	500M DUAL REGION US & EU
Subscribers	5,000,000	50,000,000	500,000,000	500,000,000
Total TPS Achieved	5,000	50,000	500,000	1,000,000
Totogi Engine CPU cores (Graviton/ARM)	48	480	4,800	9,600
Totogi Engine Memory (GBs)	96	960	9,600	19,200

Summary

This benchmark set out to demonstrate that Totogi, based on native public cloud technology, is a true webscale platform that can linearly scale to accommodate all the subscribers / devices in the world and process the associated number of transactions that are generated. This can all be done at a fraction of the cost of a traditional on-premise system while still maintaining complete data integrity and resilience.

Unlike traditional systems that have been migrated to the cloud, Totogi demonstrated that a single database instance could be used to support prepaid, postpaid, and sharing models, thus eliminating the restrictions that other vendors face when trying to scale their legacy systems in the cloud.

Totogi

For further information: