LOVELY PROFESSIONAL UNIVERSITY

Transforming Education Transforming India

Title : URL Shortener

- Name = Amar
- Registration ID = 12412085
- Roll No. = 07

LinkSwift

Multi-Region URL Shortener with Click Analytics at Scale

1. Requirements Pack

1.1 Stakeholder Analysis & Prioritization

The success of LinkSwift is measured by meeting diverse expectations across multiple user groups.

Stakeholder	Primary Concerns	Priority
End Users	Click reliability, near-zero latency, security (no malicious redirects).	High
Marketers	Link creation, customizable vanity links, reliable, nearreal-time analytics reporting (Geo, Referrer).	High
App Developers	Robust, high-throughput REST API for bulk creation and management, clear documentation.	Medium
Analytics Consumers	Access to granular, aggregated data for custom business intelligence (OLAP capability).	Medium

1.2 Functional Requirements (FRs)

Link Creation: Authenticated users must be able to submit a long URL and receive a unique, collision-free short hash (6-8 alphanumeric characters).

Customization: Users can optionally request a specific, custom hash (vanity URL).

Link Resolution: A GET request to the short URL must result in a HTTP 302 Temporary Redirect to the destination URL.

Link Expiry: Links must support optional TTL (Time-to-Live) configuration, automatically becoming non-resolvable after expiry.

Analytics Tracking: Every successful resolution must asynchronously record click metadata: timestamp, IP address (for geo-lookup), User-Agent, and Referrer header.

Analytics Retrieval: Authenticated users can query and view aggregated statistics (total clicks, top 5 geo, top 5 referrers).

1.3 Non-Functional Requirements (NFRs)

The non-functional requirements dictate the core architectural choices, prioritizing speed and durability.

Availability - Redirect: 99.99% uptime for the read path (link resolution). This is the highest priority SLO.

Latency - Redirect: p95 latency must be less than 50ms globally, achieved through geoproximity routing.

Scalability - Read: The system must scale to handle **10 Billion+ redirects per month**, with peak traffic modeled at 40,000 QPS (Queries Per Second).

Scalability - Write: The system must withstand **write bursts** up to 4,000 QPS for short periods (e.g., during mass link creation campaigns).

Data Consistency: Strong Consistency is required for link mapping data; **Eventual Consistency** is acceptable for analytics data.

Security: The system must employ client and user-based rate limiting to prevent abuse (DDoS and link flooding).

Multi-Region: The infrastructure must be deployed in at least three major global regions (e.g., US-East, EU-West, AP-Southeast) to satisfy NFR2 and NFR1.

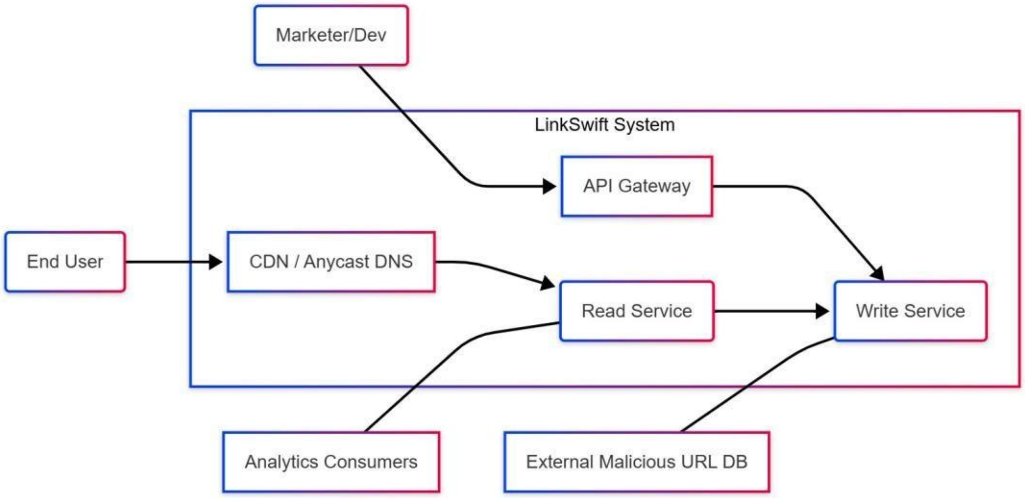
1.4 Constraints & Assumptions

Constraint: Hash generation uses Base62 encoding over a 62-bit unique ID.

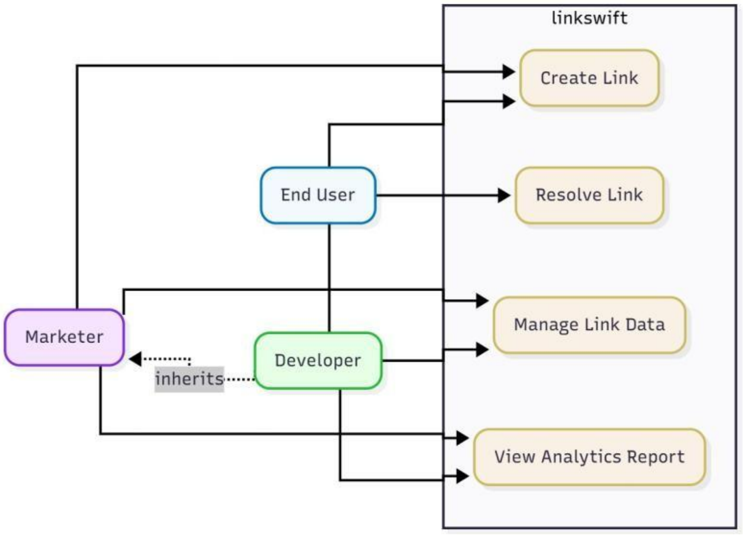
Assumption: Malicious URL scanning occurs asynchronously post-creation.

2. Diagrams & Architecture

2.1 System Context Diagram (Minimalist Flow)



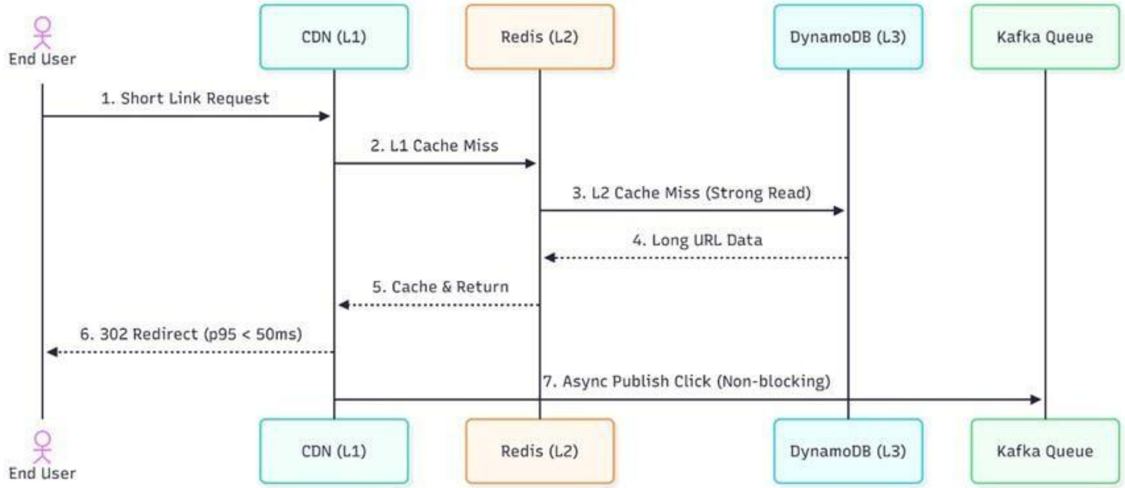
2.2 Use - Case Diagram



2.3 Core Sequence Diagrams (Flows)

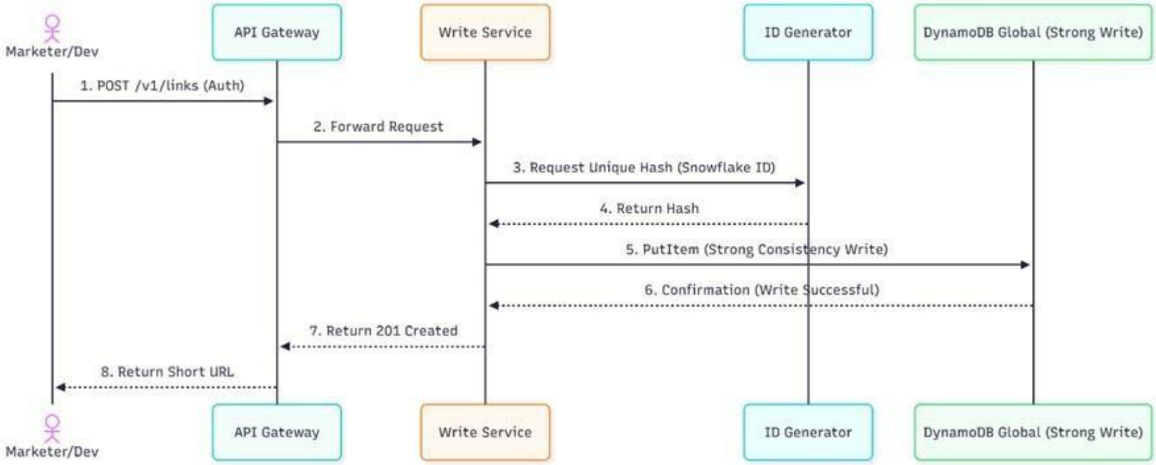
2.3.1. Read Path (The Hot Path)

The sequence must be non-blocking, prioritized for redirect latency (< 50ms).



2.3.2. Write Path (Link Creation)

This path enforces strong consistency for the new link mapping.



2.3 High-Level Architecture Components

The architecture is logically segmented into three planes: Global Edge, Application Core (Regional), and Analytics (Async).

Global Edge: Anycast DNS (low-latency routing) and CDN (L1 Caching).

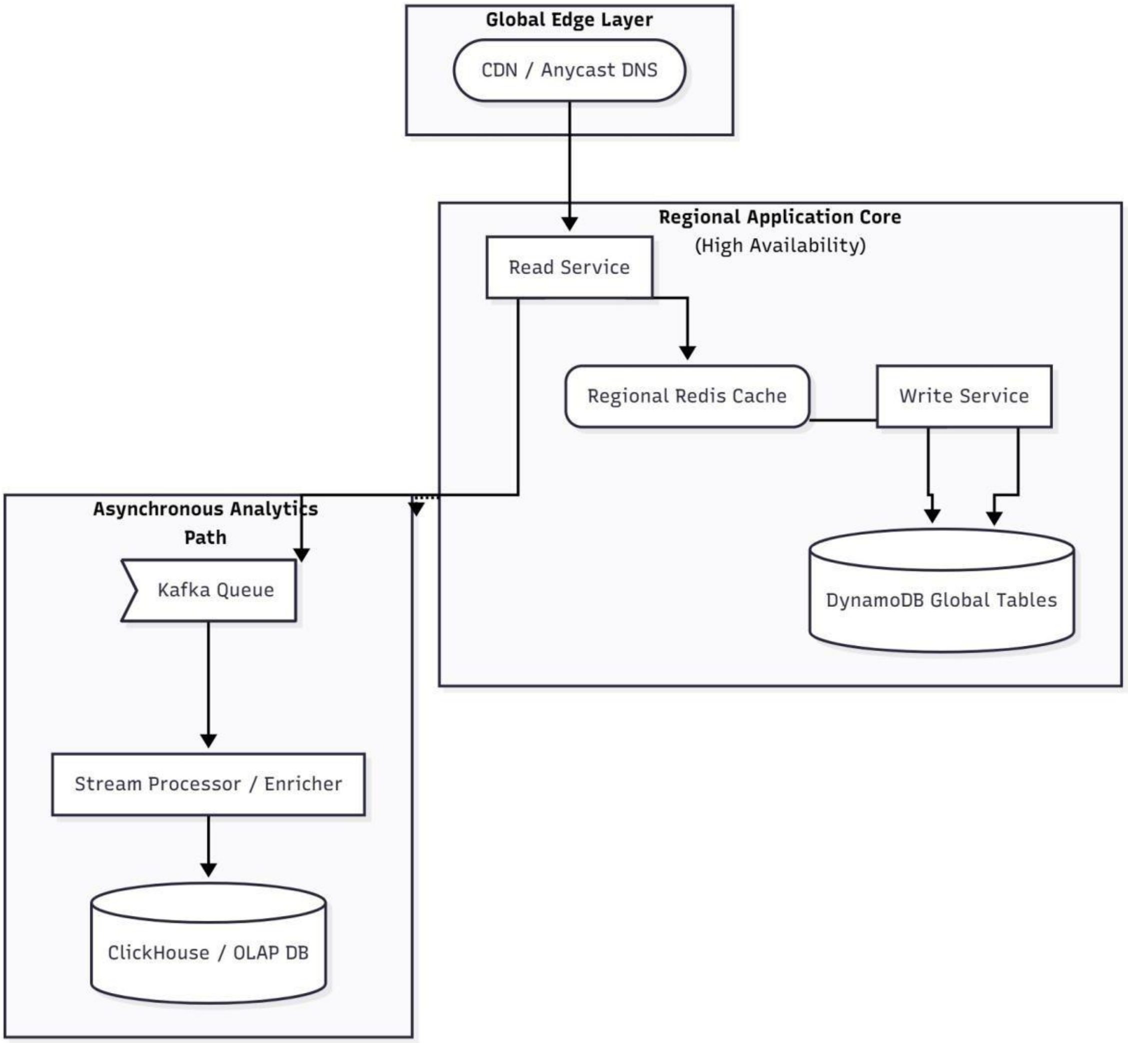
Application Core:

- **Read Service:** Highly scaled, stateless, lightweight microservice (Go or Lambda).
- **Write Service:** Handles business logic and strong consistency transactions.
- **Regional Cache (L2):** Managed Redis cluster in each region, dedicated to link mapping.

Data Stores:

- **Global Database:** DynamoDB Global Tables (multi-master NoSQL) for link mapping.
- **Analytics Queue:** Kafka or Kinesis (high-throughput, durable stream buffer).
- **Analytics DB:** ClickHouse or Redshift (OLAP system for complex, fast analytical queries).

Analytics Pipeline: Stream Processor (e.g., Flink) handles geo-enrichment and batched writes to the Analytics DB.



3. Engineering Notes and Specifications

1. Capacity Planning & Back-of-the-Envelope Sizing

The sizing strategy is built to handle massive scale by shifting the majority of the workload to caching layers.

- **Read Load Peak:** The system is sized to handle a peak load of $\approx \mathbf{38,580 \text{ Queries Per Second (QPS)}}$.
- **Cache Strategy Goal:** A combined **cache hit ratio of $\mathbf{> 99\%}$** is mandatory across all layers (CDN and Redis).
- **Database Load:** This cache strategy limits the load on the database to a manageable $\approx \mathbf{386 \text{ QPS}}$.
- **Data Volume:** The primary data challenge is **Analytics Data**, which generates $\approx \mathbf{5 \text{ TB/month}}$, requiring a dedicated OLAP solution.

2. API Specifications and Data Model

API Specifications (Core)

- **Link Creation (POST /v1/links):** Requires authentication and triggers the **Strong Consistency** write path.
- **Analytics Retrieval (GET /v1/links/{hash}/stats):** Queries the **Eventual Consistency** OLAP database for aggregate reports.

Data Model

- **Links Table (DynamoDB):** Optimized for $\mathbf{O(1)}$ lookup latency; uses the short **hash** as the Primary Key (PK).
- **Analytics Table (OLAP/ClickHouse):** Optimized for high-volume ingestion and fast reporting; clustered by **(hash, timestamp)**.

3. Consistency, Caching, and Indexing

These layers are critical for hitting the speed and availability of SLOs.

Data Consistency

- **Strong Consistency:** Used for **link mapping** (DynamoDB Global Tables) to ensure a link works instantly upon creation.
- **Eventual Consistency:** Used for **analytics data** (Kafka/OLAP), as a short delay in reporting is acceptable.

Caching and Indexing

- **L1 Cache (CDN):** Caches the final 302 Redirect at the edge (hottest layer).
- **L2 Cache (Regional Redis):** Provides the second line of defense, caching hash: long_url lookups.
- **Indexing:** The Links Table relies on the $\mathbf{PK (hash)}$ for quick reads; **GSI on user_id** supports administrative listing.

4. Resiliency and Rate Limiting

Mechanisms to ensure the **99.99% availability** SLO is never breached.

Resiliency (Retries, Timeouts, Circuit Breakers)

- **Circuit Breaker:** Used on the Read Service's connection to Kafka. If Kafka fails, the circuit opens, allowing the redirect to succeed without delay.
- **Timeouts: Aggressive timeouts** are enforced on the Read Path (e.g., Redis $<\text{10ms}$) to prevent cascading latency.
- **Retries: Disabled** on the Read Path to protect the sub-50ms latency SLO.

Rate Limiting

- **Write Path:** Limits authenticated users by **User ID** (e.g., 100 links/hour) to prevent abuse.
- **Read Path:** Limits anonymous traffic by **IP Address** (e.g., 500 redirects/minute) to mitigate DDoS attacks.

5. Observability and Maintenance Plan

Observability (Logs, Metrics, Traces)

- **Metrics:** Track the core metrics against SLOs, including $\mathbf{redirect\_p95\_latency\_ms}$, $\mathbf{cache\_hit\_ratio}$, and $\mathbf{analytics\_queue\_depth}$.
- **Traces:** Distributed tracing ($\mathbf{OpenTelemetry}$) tracks requests across multiple services for debugging performance bottlenecks.

Maintenance Plan

- **Deployments:** Use containerization to enable **blue/green, zero-downtime deployments**.
- **Data Lifecycle:** Automated policy handles the archival of old analytics data from the OLAP database to cold storage.
    ○

5. Resiliency and Quality Targets

5.1 Multi-Region Strategy and Failover

The architecture is **Active-Active** globally, utilizing Anycast DNS and DynamoDB Global Tables for automatic, low-latency redirection and data replication across regions.

5.2 Resiliency Mechanisms

- **Rate Limiting:** Implemented on both the Write Path (by user/API key) and the Read Path (by IP address) to mitigate DDoS and write bursts.
- **Circuit Breaker:** Used on the Read Service's asynchronous call to the Kafka Queue. If the queue fails, the circuit opens, guaranteeing the core redirect functionality remains 100% available.

- **Timeouts and Retries:** Aggressive timeouts are enforced on the Read Path to protect latency.

5.3 Quality Targets: Service Level Objectives (SLOs)

Metric	Target	Rationale
Redirect Availability (NFR1)	99.99%	Mission-critical: Allows for only 52 minutes of downtime per year.
Redirect Latency (NFR2)	p95 < 50ms	Core competitive advantage: Ensures instant global experience.
Analytics Freshness	99% within 15 minutes	Balances analytics immediacy with redirect performance.

5.4 Trade-Off Discussion (Final Summary)

1. **Latency vs. Consistency (Analytics):** We traded **Strong Consistency** for analytics data to guarantee the < **50ms Latency SLO** for the user.

2. **Cost vs. Performance/Resiliency:** We chose a **Multi-Region, Active-Active** architecture using expensive managed services to meet global latency and 99.99% availability, accepting the higher operational cost.