**Latency in System Design**

**📌 What is Latency?**

🔹 **Latency** is the **time delay** between a request and the corresponding response.  
🔹 It is **measured in milliseconds (ms)** and is a key performance metric for any distributed system.

**🛠️ Example: Google Search**

1️⃣ A user **searches "Latest News"** on Google.  
2️⃣ The request travels through **multiple servers** (DNS, Load Balancers, Search Index, Caching, etc.).  
3️⃣ Google fetches the results and **returns them in ~100ms**.  
4️⃣ This **100ms delay** is called **latency**.

**🔴 Why Does High Latency Occur?**

🚫 **Network Issues** → Slow internet, packet loss, long routing paths.  
🚫 **Database Bottlenecks** → Slow queries, high traffic, or poorly indexed data.  
🚫 **Backend Processing Delays** → Complex computations or overloaded servers.  
🚫 **Cross-Region Communication** → Requests going across continents introduce **extra delay**.

**🟢 How to Reduce Latency?**

**1️⃣ Caching (Reduce Response Time)**

✅ Use **Redis/Memcached** to store frequently requested data.  
✅ Example: **Facebook** caches user profiles, so profile loads **instantly**.

**2️⃣ Load Balancing (Distribute Load)**

✅ Requests are sent to the **nearest and least busy** server.  
✅ Example: **Netflix** uses a **load balancer** to distribute video streams efficiently.

**3️⃣ Content Delivery Network (CDN)**

✅ **CDNs store static content** (images, CSS, videos) on **edge servers worldwide**.  
✅ Example: **YouTube** stores videos in **CDNs** so users don’t experience buffering.

**4️⃣ Database Optimization**

✅ **Indexing** speeds up queries by avoiding full-table scans.  
✅ **Read Replicas** ensure queries are handled in parallel.  
✅ Example: **Amazon** uses **read replicas** to speed up product search.

**5️⃣ Edge Computing (Process Closer to User)**

✅ Moves computation **closer** to users instead of central data centers.  
✅ Example: **Cloudflare** processes security rules at edge servers to block attacks **before they reach the origin server**.

**📌 Latency in Different Systems**

| **System** | **Source of Latency** | **Solution** |
| --- | --- | --- |
| **Google Search** | Query processing, ranking algorithms | Caching, CDNs, AI-based query prediction |
| **Netflix Streaming** | Video buffering, network delays | CDNs, Adaptive Bitrate Streaming |
| **Amazon Product Search** | Database queries, high traffic | Read replicas, indexing |
| **Gaming (Fortnite, Call of Duty)** | Network lag, server delays | Edge computing, UDP for faster packet transmission |

**🛠️ Final Thoughts: When to Optimize Latency?**

✅ If **real-time processing** is needed (e.g., stock trading, gaming).  
✅ When **database queries** are too slow.  
✅ If your app **relies on frequent network requests** (e.g., social media feeds, chat apps).