CDN — [Notes]

What is a CDN

When a website has an I/O bottleneck serving static content the traditional solution is to use a CDN.

- CDN allow placing your static assets close to the users
- Reduces costs
- Decreases latency
- Increases complexity of your system
- A significant chunk of resources served on a webpage may not be changing on an hourly or daily basis in those cases we'll cache these at the CDN level
- CDNS not only help reduce the load on our servers by removing the burden of serving static and bandwidth-intensive resources they also let us be present closer to our users by way of points of presence
- CDNS also let us do geo load balancing in case we have multiple data centers around the world

Static Content

- Images
- CSS
- HTML
- JavaScript

Type of CDN

- **Push**
  ✅ Initiated by client
  ✅ Good when not much static content
- **Pull**
  ✅ Pulled by CDN when a user request and data doesn't exist
  ✅ Suitable when lot of static content

Is a CDN the same as a web host?

While a CDN does not host content and can't replace the need for proper web hosting, it does help cache content at the network edge, which improves website performance. Many websites struggle to have their performance needs met by traditional hosting services, which is why they opt for CDNs.

By utilizing caching to reduce hosting bandwidth, helping to prevent interruptions in service, and improving security, CDNs are a popular choice to relieve some of the major pain points that come with traditional web hosting.

What are the benefits of using a CDN?

Although the benefits of using a CDN vary depending on the size and needs of an Internet property, the primary benefits for most users can be broken down into 4 different components:

1. **Improving website load times** — By distributing content closer to website visitors by using a nearby CDN server (among other optimizations), visitors experience faster page loading times. As visitors are more inclined to click away from a slow-loading site, a CDN can reduce bounce rates and increase the amount of time that people spend on the site. In other words, a faster a website means more visitors will stay and stick around longer.

2. **Reducing bandwidth costs** — Bandwidth consumption costs for website hosting is a primary expense for websites. Through caching and other optimizations, CDNs are able to reduce the amount of data an origin server must provide, thus reducing hosting costs for website owners.

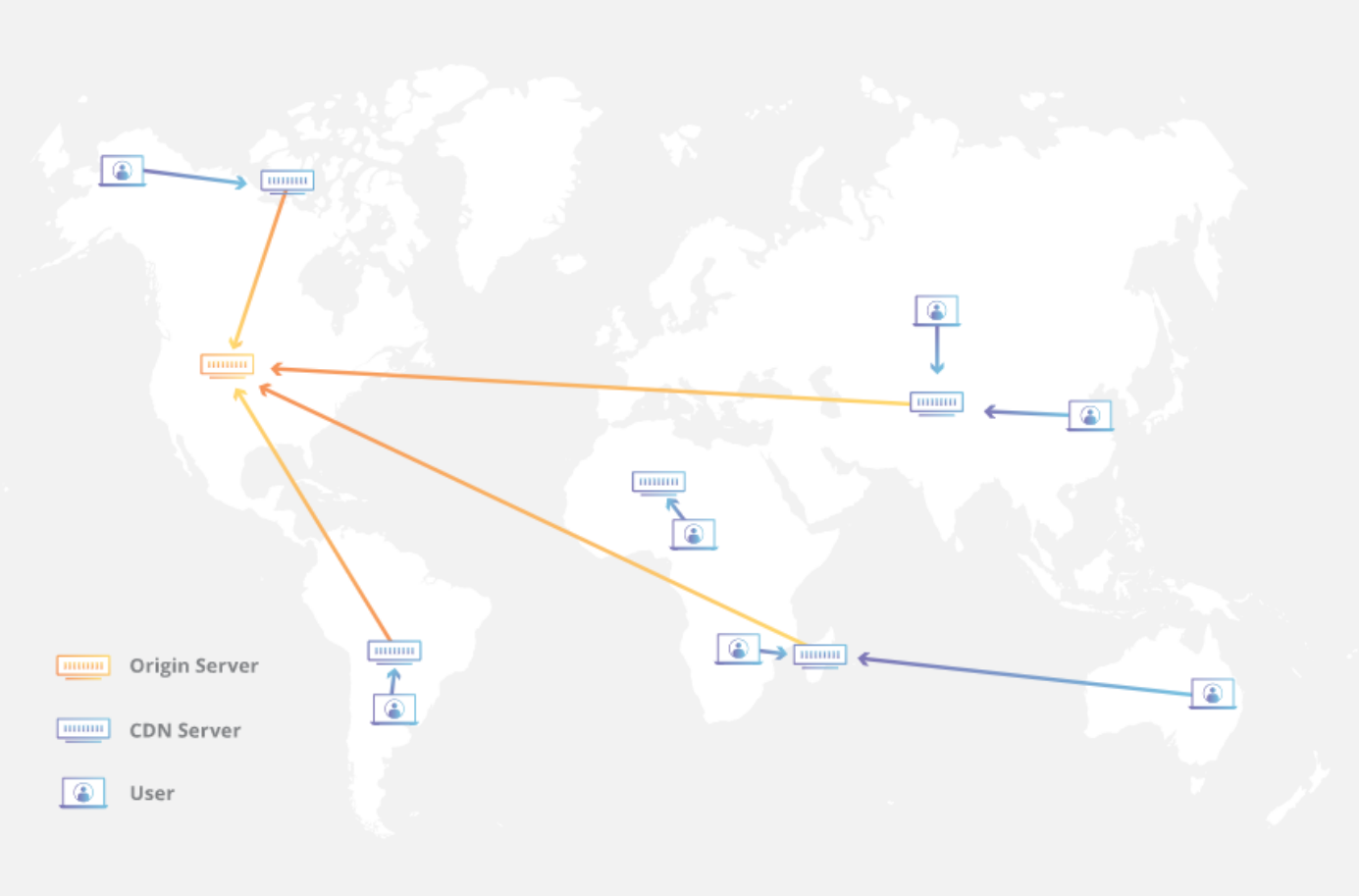
3. **Increasing content availability and redundancy** — Large amounts of traffic or hardware failures can interrupt normal website function. Thanks to their distributed nature, a CDN can handle more traffic and withstand hardware failure better than many origin servers.

4. **Improving website security** — A CDN may improve security by providing DDoS mitigation, improvements to security certificates, and other optimizations.

How does a CDN work?

At its core, a CDN is a network of servers linked together with the goal of delivering content as quickly, cheaply, reliably, and securely as possible. In order to improve speed and connectivity, a CDN will place servers at the exchange points between different networks.

These Internet exchange points (IXPs) are the primary locations where different Internet providers connect in order to provide each other access to traffic originating on their different networks. By having a connection to these high speed and highly interconnected locations, a CDN provider is able to reduce costs and transit times in high speed data delivery.



Beyond placement of servers in IXPs, a CDN makes a number of optimizations on standard client/server data transfers. CDNs place Data Centers at strategic locations across the globe, enhance security, and are designed to survive various types of failures and Internet congestion.

Typical CDN Flow

Below is the example from the Haystack paper [this is the old flow i.e. prior to haystack but still holds good for typical CDN flow]

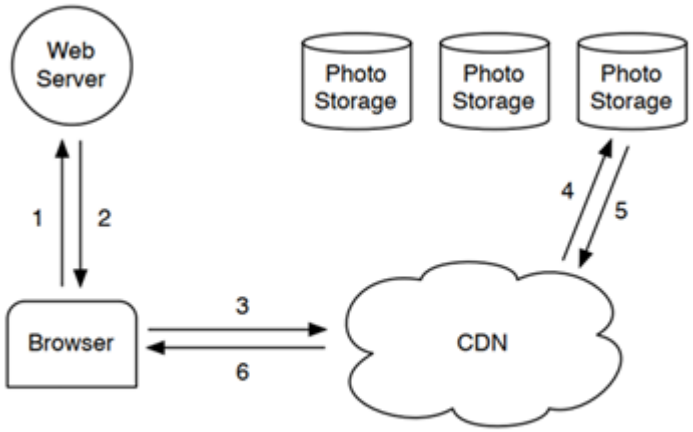


Figure 1: Typical Design

Step 1: a user visits a page containing an image until she downloads that image from its location on disk. When visiting a page the user's browser first sends an HTTP request to a web server

Step 2: web server is responsible for generating the markup for the browser to render.

Step 3: For each image, the web server constructs a URL directing the browser to a location from which to download the data. For popular sites, this URL often points to a CDN.

Step 4: If the CDN has the image cached then the *CDN responds immediately with the data. (**Step 6**)*

Step 5: Otherwise, the CDN examines the URL, which has enough information embedded to retrieve the photo from the site's storage systems.

Step 6: The CDN then updates its cached data and sends the image to the user's browser

---

*major lesson we learned is that CDNs by themselves do not offer a practical solution to serving photos on a social networking site. CDNs do effectively serve the hottest photos — profile pictures and photos that have been recently uploaded — but a social networking site like Facebook also generates a large number of requests for less popular (often older) content, which we refer to as the **long tail**.*

*Requests from the long tail account for a significant amount of our traffic, almost all of which accesses the backing photo storage hosts as these requests typically **miss in the***

CDN

Latency — How does a CDN improve website load times?

When it comes to websites loading content, users drop off quickly as a site slows down. CDN services can help to reduce load times in the following ways:

- The globally distributed nature of a CDN means reduce distance between users and website resources. Instead of having to connect to wherever a website's origin server may live, a CDN lets users connect to a geographically closer data center. Less travel time means faster service.
- Hardware and software optimizations such as efficient load balancing and solid-state hard drives can help data reach the user faster.
- CDNs can reduce the amount of data that's transferred by reducing file sizes using tactics such as minification and file compression. Smaller file sizes mean quicker load times.
- CDNs can also speed up sites which use TLS/SSL certificates by optimizing connection reuse and enabling TLS false start.