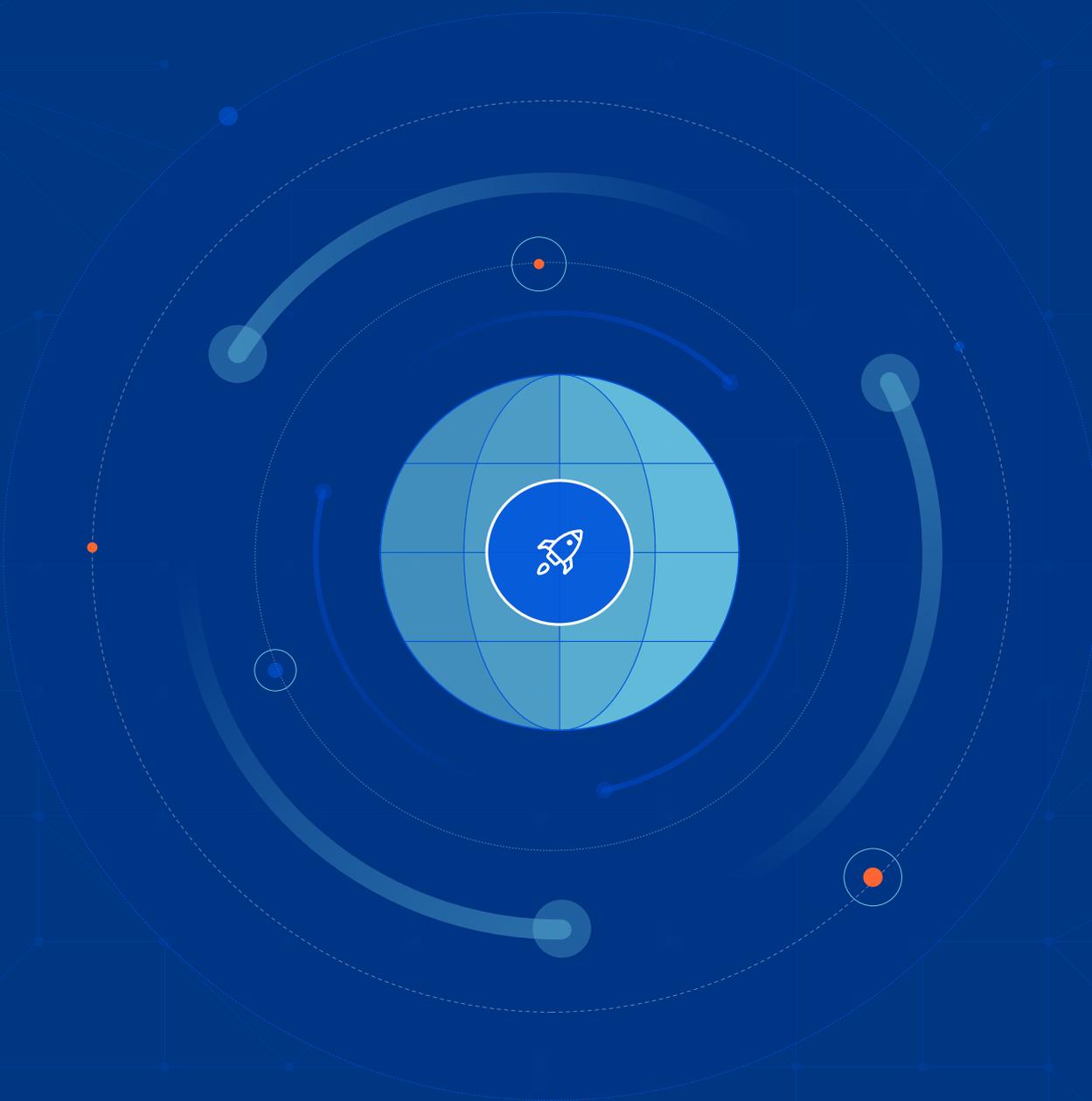


# Cloudflare CDN- Referenzarchitektur



# INHALT

---

Klicken, um zum jeweiligen Abschnitt zu springen

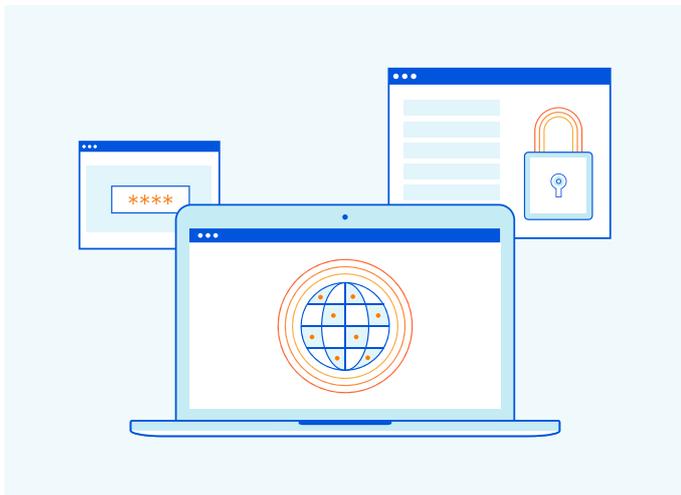
<b>Übersicht</b> .....	<b>3</b>
<b>Traditionelle Herausforderungen bei der Bereitstellung von Webanwendungen</b> .....	<b>4-5</b>
<b>Wie ein CDN die Herausforderungen von Webanwendungen meistert</b> .....	<b>6</b>
<b>Einführung in das Cloudflare CDN</b> .....	<b>7</b>
<b>Cloudflare CDN-Architektur und -Design</b> .....	<b>8-9</b>
Argo Tiered Cache .....	9
Cloudflare Tiered Cache – Topologien .....	10
Traffic-Fluss: Argo Tiered Cache, Smart Tiered Cache-Topologie .....	11-12
Argo Smart Routing .....	12
Traffic-Fluss: Argo Tiered Cache, Smart Tiered Cache-Topologie mit Argo Smart Routing .....	13-14
<b>Zusammenfassung</b> .....	<b>15</b>

# Übersicht

---

Jeden Tag profitieren Internetnutzer von den Vorteilen der Performance und Zuverlässigkeit von Content Delivery Networks (CDNs). CDNs sind zu einem Muss geworden, um Latenzzeiten zu bekämpfen. Jedes größere Unternehmen, das Inhalte für Benutzer im Internet bereitstellt, benötigt ein CDN. CDNs bieten Kunden nicht nur Performance und Zuverlässigkeit. Sie ermöglichen es Unternehmen auch, ihre Anwendungen besser abzusichern und Kosten zu senken. In diesem Dokument werden die traditionellen Herausforderungen erörtert, mit denen Kunden bei Webanwendungen konfrontiert sind, wie das Cloudflare CDN diese Herausforderungen löst und wie CDN-Architektur und -Design aussehen.

# Traditionelle Herausforderungen bei der Bereitstellung von Webanwendungen



In den letzten Jahren, insbesondere angesichts der COVID-19-Pandemie und der Konzentration auf Remote-Arbeit, hat der Internet-Traffic erheblich zugenommen. Dadurch ist es noch wichtiger geworden, den Netzwerk-Traffic effizient zu verwalten, die Latenz zu verringern und die Performance zu steigern.

### Unternehmen, die ihre Anwendungen in der Cloud oder vor Ort betreiben, stehen vor den folgenden Herausforderungen:

1. Implementierung von Lösungen, welche die Performance steigern
2. Skalierung der Architektur bei steigender Nachfrage, um Verfügbarkeit und Redundanz zu gewährleisten
3. Schutz ihrer Umgebungen und Anwendungen vor den wachsenden Bedrohungen aus dem Internet
4. Eindämmung der steigenden Kosten im Zusammenhang mit all den oben genannten Tätigkeiten

Bei Unternehmen, die Kunden auf der ganzen Welt haben, sind die oben genannten Herausforderungen ein großes Unterfangen. Traditionell wird eine Website/Anwendung zentral bereitgestellt und aus Gründen der Verfügbarkeit in eine andere Region repliziert. Oder die Website/Anwendung wird auf einer begrenzten Anzahl von Servern bereitgestellt, manchmal über mehrere Rechenzentren verteilt, um Ausfallsicherheit zu gewährleisten.

Die Server, auf denen die Websites gehostet werden, heißen Ursprungsserver. Wenn Clients auf eine Website zugreifen, stellen sie eine Anfrage nach Ressourcen vom Server. Die Navigation auf einer Website kann Hunderte von Anfragen des Browsers für HTML, CSS, Bilder, Videos usw. erzeugen. Bei HTTP-Versionen vor HTTP/2 würde jede dieser HTTP-Anfragen auch eine neue TCP-Verbindung erfordern.

Verbesserungen in HTTP/2 ermöglichen das Multiplexing mehrerer Anfragen an denselben Server über eine einzige TCP-Verbindung, wodurch Serverressourcen eingespart werden. Allerdings werden immer noch Datenverarbeitungs- und Netzwerkressourcen verbraucht, wenn Server auf diese Anfragen reagieren. Wenn mehr Kunden auf die Website zugreifen, kann sich Folgendes ergeben:

- Der Ursprungsserver wird mit Anfragen überlastet, was sich auf die Verfügbarkeit auswirkt; Unternehmen beginnen, eine Skalierung zu erwägen, um die zusätzliche Last zu bewältigen.
- Da jede Anfrage den Weg zum Ursprungsserver zurücklegen muss, werden Performance und Benutzerfreundlichkeit durch die Latenzzeit beeinträchtigt.
- Die Latenzzeit für Endbenutzer ist proportional zur Entfernung zwischen dem Client und dem Ursprungsserver. Dies führt zu unterschiedlichen Erfahrungen je nach Standort des Clients.
- Ursprungsserver antworten auf die zunehmenden Anfragen, wodurch Bandbreiten-, Egress- und Datenverarbeitungskosten drastisch steigen.
- Selbst wenn die Kunden die Skalierung vornehmen, um die steigende Nachfrage nach Datenverkehr zu bewältigen, sind sie sowohl auf Infrastruktur- als auch auf Anwendungsebene verteilten Denial-of-Service-Angriffen (DDoS) ausgesetzt.

## Traditionelle Herausforderungen bei der Bereitstellung von Webanwendungen (Fortsetzung)

In Abbildung 1 unten ist kein CDN vorhanden und der Ursprungsserver steht in den USA. Clients greifen auf die Webseite zu. Der erste Schritt ist die DNS-Auflösung, die in der Regel vom ISP des Benutzers vorgenommen wird. Der nächste Schritt ist die HTTP-Anfrage, die direkt an den Ursprungsserver gesendet wird. Das Nutzererlebnis wird je nach Standort unterschiedlich sein. Sie können zum Beispiel sehen, dass die Latenzzeit für Benutzer in den USA, wo sich der Ursprungsserver befindet, viel geringer ist. Für Benutzer außerhalb der USA erhöht sich die Latenzzeit, was zu einer höheren Paketumlaufzeit (Round-Trip-Time, RTT) führt.

Je mehr Clients Anfragen an den Ursprungsserver stellen, desto höher ist die Belastung des Netzwerks und des Servers. Dies führt zu höheren Latenzzeiten und höheren Kosten für die Nutzung von Ressourcen und Bandbreite.

Aus dem Blickwinkel der Sicherheit ist der Ursprungsserver auch anfällig für DDoS-Angriffe sowohl auf der Infrastruktur- als auch auf der Anwendungsebene. Ein DDoS-Angriff könnte von einem Botnet initiiert werden, das Millionen von Anfragen an den Ursprungsserver sendet, wodurch Ressourcen verbraucht werden und der Server keine legitimen Kunden mehr bedienen kann.

Wenn der Ursprungsserver vorübergehend offline geht, sind alle Inhalte für die Benutzer unzugänglich.

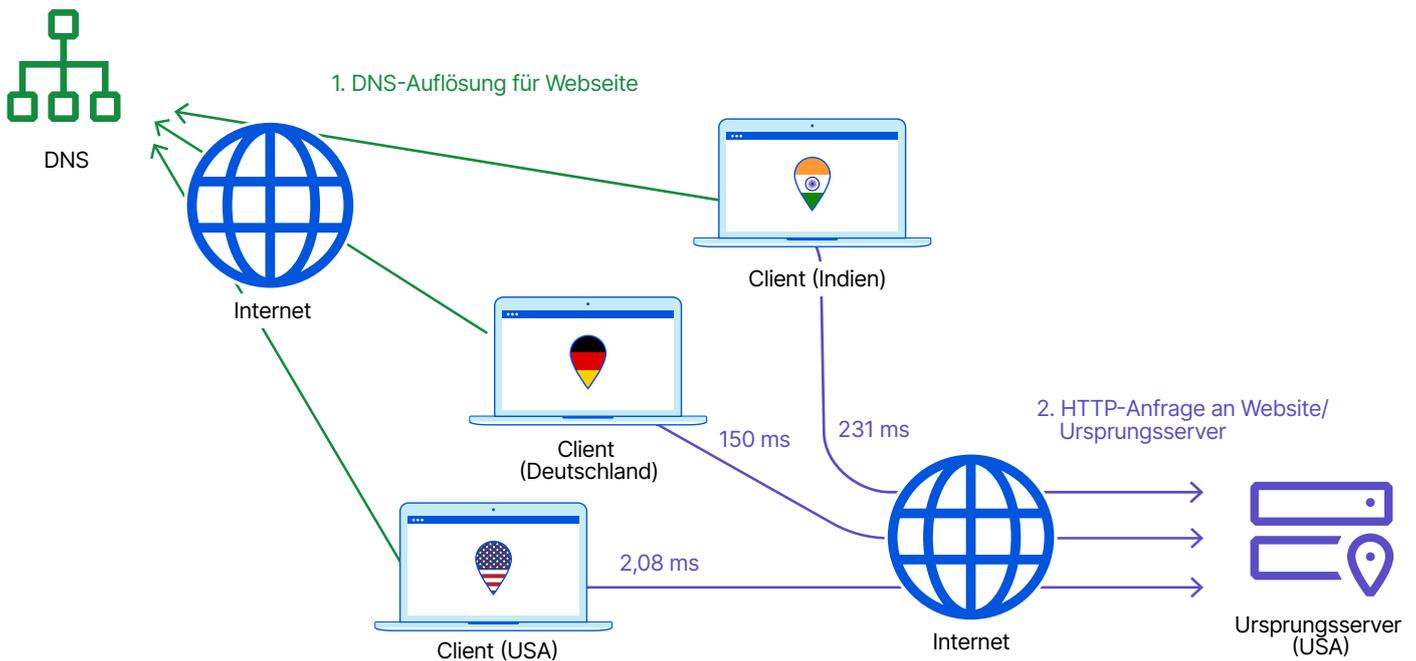


Abbildung 1: HTTP-Anfrage ohne CDN

## Wie ein CDN die Herausforderungen von Webanwendungen meistert

Ein CDN hilft jene Herausforderungen zu lösen, denen Kunden in Bezug auf Latenz, Performance, Verfügbarkeit, Redundanz, Sicherheit und Kosten gegenüberstehen. [Das Hauptziel eines CDN ist es, die Latenzzeit zu verringern und die Performance für Websites und Anwendungen zu erhöhen.](#) Dazu werden die Inhalte so nah wie möglich an den Endnutzern oder denjenigen, die auf die Inhalte zugreifen, zwischengespeichert.

CDNs verringern die Latenzzeit und erhöhen die Performance, indem sie über viele Rechenzentren auf der ganzen Welt verfügen, die die Inhalte vom Ursprung aus in den Cache stellen. Das Ziel ist es, die Inhalte so nah wie möglich an den Nutzern zwischenzuspeichern. Daher werden die Inhalte am Netzwerkrand des CDN-Anbieters zwischengespeichert.

Die Auswirkungen sind:

- **Verbesserte Ladezeit von Websites**  
Anstatt dass jeder Client eine Anfrage an den Ursprungsserver stellt, der sich möglicherweise in beträchtlicher Entfernung befindet, wird die Anfrage an einen lokalen Server weitergeleitet, der mit zwischengespeicherten Inhalten antwortet. Dadurch werden die Latenzzeiten verringert und die Performance insgesamt erhöht. Unabhängig davon, wo sich der Ursprungsserver und die Clients befinden, ist die Performance für alle Benutzer einheitlicher, da das CDN nach Möglichkeit lokal zwischengespeicherte Inhalte bereitstellt.
- **Erhöhte Verfügbarkeit und Redundanz von Inhalten**  
Da nicht mehr jede Client-Anfrage an den Ursprungsserver gesendet werden muss, bieten CDNs nicht nur Vorteile bei der Performance, sondern auch bei der Verfügbarkeit und Redundanz. Die Anfragen werden auf lokale Server mit zwischengespeicherten Inhalten verteilt. Diese Server beantworten lokale Anfragen, wodurch die Gesamtlast auf dem Ursprungsserver erheblich verringert wird. Der Ursprungsserver wird nur bei Bedarf kontaktiert (wenn Inhalte nicht zwischengespeichert werden oder bei dynamischen Inhalten, die nicht zwischengespeichert werden können).
- **Verbesserte Websitesicherheit**  
Ein CDN fungiert als Reverse Proxy und ist den Ursprungsservern vorgelagert. Dadurch kann es mehr Sicherheit bieten, z. B. DDoS-Abwehr, Verbesserungen bei Sicherheitszertifikaten und andere Optimierungen.
- **Reduzierte Bandbreitenkosten**  
Da CDNs zwischengespeicherte Inhalte verwenden, um auf Anfragen zu antworten, wird die Anzahl der an den Ursprungsserver gesendeten Anfragen reduziert. Dadurch sinken auch die damit verbundenen Bandbreitenkosten.

Ein wichtiger Unterschied bei einigen CDN-Implementierungen besteht darin, wie sie den Traffic zu den jeweiligen lokalen CDN-Knoten leiten.

Die Weiterleitung von Anfragen an CDN-Knoten kann über zwei verschiedene Methoden erfolgen:

### 1. DNS-Unicast-Routing

Bei dieser Methode werden rekursive DNS-Anfragen an CDN-Knoten umgeleitet; der DNS-Resolver des Clients leitet Anfragen an den autoritativen Nameserver des CDN weiter. CDNs, die auf DNS-Unicast-Routing basieren, sind nicht ideal, da die Clients geografisch weit vom DNS-Resolver entfernt sein können. Die Entscheidung über die nächstgelegenen CDN-Knoten basiert auf dem DNS-Server des Kunden und nicht auf der IP-Adresse des Kunden.

Wenn Änderungen für die DNS-Antwort erforderlich sind, besteht außerdem eine Abhängigkeit vom Ablauf der DNS-Time-to-Live (TTL).

Da das DNS-Routing Unicast-Adressen verwendet, wird der Datenverkehr direkt zu einem bestimmten Knoten geleitet, was zu Problemen führen kann, wenn es zu Traffic-Spitzen kommt, wie z. B. bei einem DDoS-Angriff.

Eine weitere Herausforderung bei DNS-basierten CDNs besteht darin, dass DNS bei einem Ausfall nicht sehr zuverlässig ist. Normalerweise muss eine neue Sitzung oder Anwendung gestartet werden, damit der DNS-Resolver mit einer anderen IP-Adresse übernehmen kann.

### 2. Anycast-Routing

Das Cloudflare CDN, auf das im nächsten Abschnitt näher eingegangen wird, verwendet Anycast-Routing. Anycast ermöglicht es, dass Knoten in einem Netzwerk die gleiche IP-Adresse haben. Dieselbe IP-Adresse wird von mehreren Knoten an verschiedenen Standorten bekannt gegeben, und die Umleitung der Clients erfolgt über das Routing-Protokoll des Internets, BGP.

Die Verwendung eines Anycast-basierten CDN hat mehrere Vorteile:

- Eingehender Traffic wird an das nächstgelegene Rechenzentrum weitergeleitet, das über die nötige Kapazität zur effizienten Verarbeitung der Anfragen verfügt.
- Verfügbarkeit und Redundanz sind von Haus aus gegeben. Da mehrere Knoten die gleiche IP-Adresse haben, werden bei einem Ausfall eines Knotens die Anfragen einfach an einen anderen Knoten in der Nähe weitergeleitet.
- Da Anycast den Traffic auf mehrere Rechenzentren verteilt, vergrößert es die Gesamtfläche und verhindert so, dass ein einzelner Standort mit Anfragen überlastet wird. Aus diesem Grund sind Anycast-Netzwerke sehr widerstandsfähig gegenüber DDoS-Angriffen.

## Wir präsentieren: Cloudflare CDN

Cloudflare bietet ein Software-as-a-Service-Modell (SaaS) für CDN. Mit dem SaaS-Modell von Cloudflare profitieren die Kunden vom Cloudflare CDN, ohne dass sie eine Infrastruktur oder Software verwalten oder unterhalten müssen.

Die Vorteile des Cloudflare CDN lassen sich auf die beiden folgenden Punkte zurückführen, die in diesem Abschnitt näher erläutert werden.

**1. CDNs erhöhen von Natur aus die Performance, indem sie Inhalte auf Servern in der Nähe des Benutzers zwischenspeichern.**

**2. Die einzigartige Cloudflare-Architektur und das integrierte Ökosystem**

Abbildung 2 zeigt eine vereinfachte Ansicht des Cloudflare CDN. Die Kunden erhalten ihre Antwort von einem Server im globalen Anycast-Edge-Netzwerk von Cloudflare, der dem Standort der Kunden am nächsten ist, wodurch die Latenzzeit und die RTT drastisch reduziert werden. Das Diagramm zeigt eine konsistente Endbenutzererfahrung unabhängig vom physischen Standort der Clients und der Herkunft.

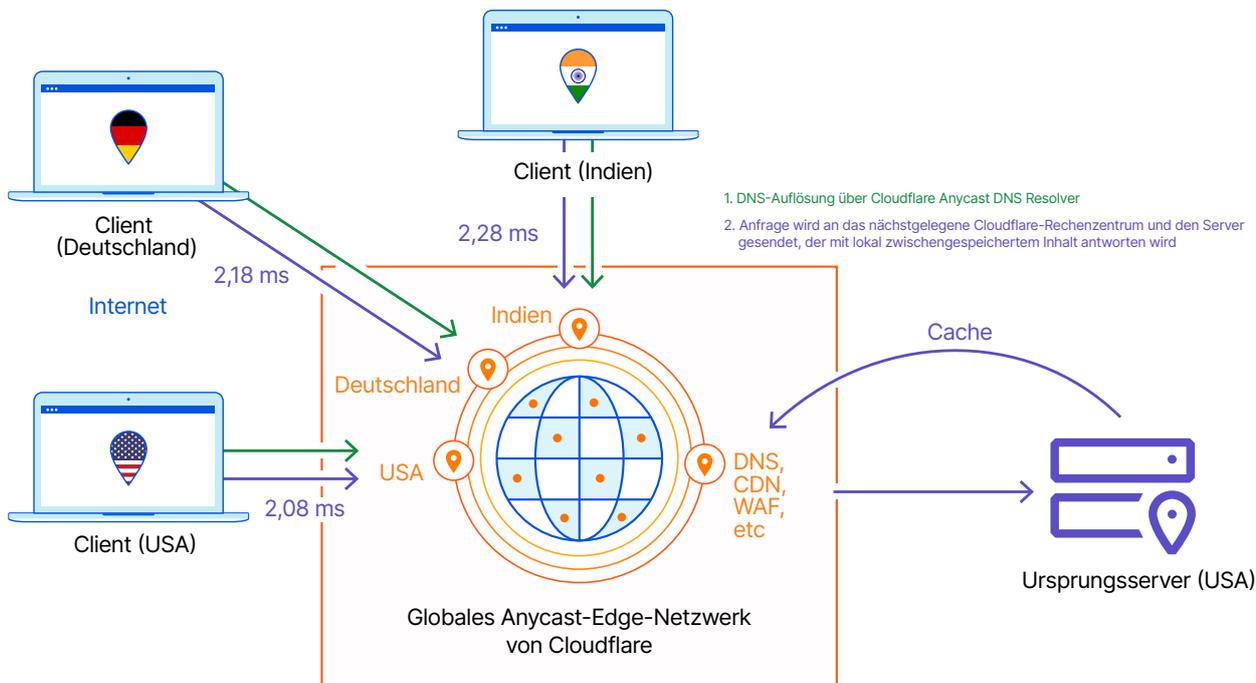


Abbildung 2: HTTP-Anfrage an Cloudflare CDN mit Anycast

# Architektur und Design des Cloudflare CDN

Abbildung 3 ist eine Ansicht des Cloudflare CDN auf dem globalen Anycast-Edge-Netzwerk. Neben der Verwendung von Anycast für die Performance und Ausfallsicherheit des Netzwerks nutzt das Cloudflare CDN den Argo Tiered Cache, um optimale Ergebnisse und gleichzeitig Kostenersparnisse für die Kunden zu erzielen. Kunden können auch Argo Smart Routing aktivieren, um den schnellsten Netzwerkpfad für die Weiterleitung von Anfragen an den Ursprungsserver zu finden. Diese Fähigkeiten werden im weiteren Verlauf dieses Dokuments im Detail erläutert.

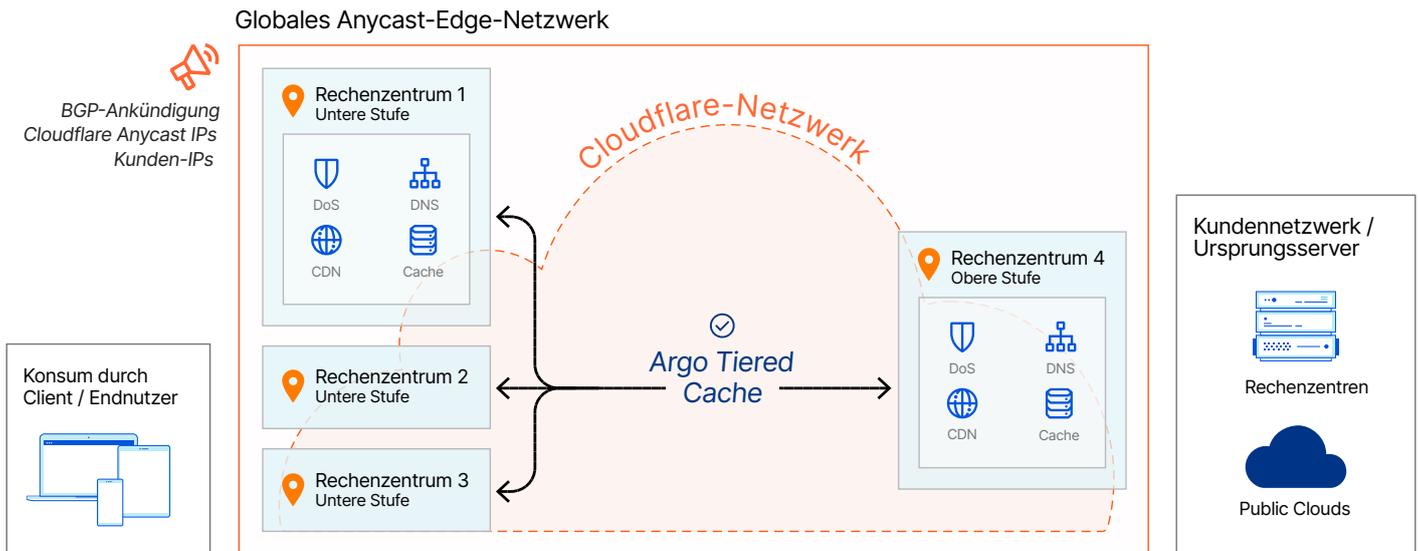


Abbildung 3: Cloudflare CDN mit Argo Tiered Cache im globalen Anycast-Edge-Netzwerk

In dem obigen Diagramm sieht man einige wichtige Schlüsselpunkte über das Cloudflare CDN und das globale Anycast-Edge-Netzwerk, in dem es sich befindet:

- Ein wichtiges Unterscheidungsmerkmal ist, dass Cloudflare ein globales Netzwerk nutzt und jeden Service auf jedem Server in jedem Cloudflare-Rechenzentrum ausführt. Dadurch erhalten die Endbenutzer die größte Nähe zu den Cloudflare-Services mit der höchsten Skalierung, Ausfallsicherheit und Performance.
- Cloudflare ist ein Reverse-Proxy, d.h. er empfängt Anfragen von Kunden und leitet die Anfragen zurück an die Ursprungsserver des Kunden.

Damit durchläuft jede Anfrage das Netzwerk von Cloudflare, bevor sie das Netzwerk des Kunden erreicht.

Da Cloudflare seine Infrastruktur am Rand (Ingress) gestärkt und geschützt hat, sind alle Kunden folglich auch vor DDoS-Angriffen auf Infrastrukturebene und vor volumetrischen Angriffen geschützt. Anfragen und Datenverkehr müssen das geschützte Cloudflare-Netzwerk durchlaufen, bevor sie den Ursprungsserver des Kunden erreichen.

- Das Cloudflare CDN nutzt das globale Anycast-Edge-Netzwerk von Cloudflare. So wird die eingehende Anfrage an den Knoten weitergeleitet und von diesem beantwortet, der dem Benutzer am nächsten ist (Eyeball).
- Die inhärenten Vorteile von Anycast sind geringere Latenzzeiten, die Ausfallsicherheit des Netzwerks, eine höhere Verfügbarkeit. Auch die Sicherheit ist höher, da eine größere Angriffsfläche sowohl für legitimen Datenverkehr als auch für DDoS-Angriffe zur Verfügung steht.

# Architektur und Design des Cloudflare CDN (Fortsetzung)

Das globale Anycast-Edge-Netzwerk von Cloudflare erstreckt sich über mehr als 250 Städte in über 100 Ländern. Es erreicht 95 % der mit dem Internet verbundenen Weltbevölkerung innerhalb von 50 Millisekunden, während es gleichzeitig 100 Tbit/s an Netzwerkkapazität und DDoS-Schutz bietet.

- Edge-Knoten innerhalb des Cloudflare-Netzwerks zwischenspeichern Inhalte vom Ursprungsserver und sind in der Lage, auf Anfragen über eine zwischengespeicherte Kopie zu antworten. Cloudflare bietet auch DNS, DDoS-Schutz, WAF und andere Performance-, Zuverlässigkeits- und Sicherheitsservices, die dieselbe Edge-Architektur nutzen.

- Argo verwendet eine optimierte Routing- und Caching-Technologie im Cloudflare-Netzwerk, um Antworten noch schneller, zuverlässiger und sicherer an die Nutzer zu übermitteln. Argo umfasst Smart Routing und Tiered Cache. Cloudflare nutzt Argo, um eine erweiterte CDN-Lösung anzubieten.

## Argo Tiered Cache

Sobald eine Website hinzugefügt wurde, ist das Standard-Caching standardmäßig konfiguriert. Beim Standard-Caching fungiert jedes Rechenzentrum als direkter Reverse-Proxy für die Ursprungsserver. Ein Cache-Fehler (Cache-Miss) in einem beliebigen Rechenzentrum führt dazu, dass eine Anfrage vom eingehenden Rechenzentrum an den Ursprungsserver gesendet wird.

Obwohl das Standard-Caching funktioniert, ist es nicht die optimale Lösung. Warum? Zwischengespeicherte Inhalte, die sich näher am Kunden befinden, können bereits in anderen Cloudflare-Rechenzentren vorhanden sein. Dadurch werden die Ursprungsserver manchmal unnötig überlastet. Daher ist es am besten, den Argo Tiered Cache zu aktivieren, der in jedem Cloudflare-Tarif enthalten ist. Mit Argo Tiered Cache fungieren bestimmte Rechenzentren für andere Rechenzentren als Reverse-Proxies zum Ursprung. Dies führt zu mehr Cache-Treffern und kürzeren Antwortzeiten.

Argo Tiered Cache nutzt die Größe des Cloudflare-Netzwerks, um die Anfragen an den Ursprungsserver des Kunden zu minimieren. Wenn eine Anfrage in einem Cloudflare-Rechenzentrum eingeht und der angeforderte Inhalt nicht lokal zwischengespeichert ist, werden andere Cloudflare-Rechenzentren auf den zwischengespeicherten Inhalt überprüft.

Die Cloudflare-Rechenzentren haben kürzere Entfernungen und schnellere Pfade zwischen ihnen als die Verbindungen zwischen den Rechenzentren und den Ursprungsservern der Kunden. Dadurch wird die Antwort an den Kunden optimiert und die Cache-Trefferquote deutlich verbessert. Das Cloudflare CDN nutzt die Argo Smart Routing-Daten, um die besten Rechenzentren der oberen Stufe für den Argo Tiered Cache zu ermitteln. Argo Smart Routing kann auch als Add-on aktiviert werden, um die schnellsten Pfade zwischen Rechenzentren und Ursprungsservern für Cache-Fehler und andere Arten von dynamischem Datenverkehr bereitzustellen.

Mit dem Cloudflare CDN können Kunden ein mehrstufiges Caching konfigurieren. Beachten Sie, dass je nach Cloudflare-Plan unterschiedliche Topologien für Argo Tiered Cache verfügbar sind. Das Tiered Caching ist standardmäßig deaktiviert und kann auf der Registerkarte Caching im Hauptmenü aktiviert werden.

## Argo Tiered Cache – Topologien

Mit den verschiedenen Cache-Topologien können Kunden steuern, wie Cloudflare mit den Ursprungsservern interagiert. Auf diese Weise können sie höhere Cache-Trefferquoten, weniger Verbindungen zu Ursprungsservern und geringere Latenzzeiten sicherstellen.

Argo Tiered Cache – Topologien		
Smart Tiered Cache-Topologie (Alle Tarife)	Generische Global Tiered-Topologie (nur Enterprise)	Benutzerdefinierte Tiered Cache-Topologie (nur Enterprise)
<ul style="list-style-type: none"> <li>• Empfohlen für die meisten Implementierungen. Dies ist die Standardkonfiguration, sobald der Tiered Cache aktiviert ist.</li> <li>• Ideal für Kunden, die das CDN für die Performance nutzen, aber die Anfragen an die Ursprungsserver und die Bandbreitennutzung zwischen Cloudflare und den Ursprungsservern minimieren möchten.</li> <li>• Cloudflare findet anhand der Performance- und Routing-Daten von Argo dynamisch die beste obere Stufe für einen Ursprungsserver.</li> </ul>	<ul style="list-style-type: none"> <li>• Empfohlen für diejenigen, die einen hohen Traffic haben, der über den ganzen Globus verteilt ist, und die die höchste Cache-Nutzung und die bestmögliche Performance wünschen.</li> <li>• Die generische Global Tiered-Topologie schafft ein Gleichgewicht zwischen Cache-Effizienz und Latenz. Weist Cloudflare an, alle Tier 1-Rechenzentren als obere Stufen zu verwenden.</li> </ul>	<ul style="list-style-type: none"> <li>• Empfohlen für Kunden, die über zusätzliche Daten zu ihrer Nutzerbasis verfügen und sich auf bestimmte geografische Regionen konzentrieren möchten.</li> <li>• Mit der benutzerdefinierten Tiered Cache-Topologie können Kunden eine benutzerdefinierte Topologie einrichten, die ihren spezifischen Anforderungen entspricht (z. B. obere Stufen an bestimmten geografischen Standorten, die mehr Kunden bedienen).</li> <li>• Wenden Sie sich an einen Customer Success Manager (CSM), um eine benutzerdefinierte Topologie zu erstellen.</li> </ul>

## Traffic-Fluss: Argo Tiered Cache, Smart Tiered Cache-Topologie

In Abbildung 4 ist Argo Tiered Caching mit Smart Tiered Cache-Topologie aktiviert. Das Diagramm zeigt zwei separate Traffic-Flüsse, die im Folgenden zusammengefasst sind. Der erste Traffic-Fluss (Client 1 in grün) ist eine Anfrage von einem Client, der im Rechenzentrum 1 ankommt. Der zweite Traffic-Flow (Client 2 in lila) ist eine nachfolgende Anfrage für dieselbe Ressource in einem anderen Rechenzentrum, Rechenzentrum 2.

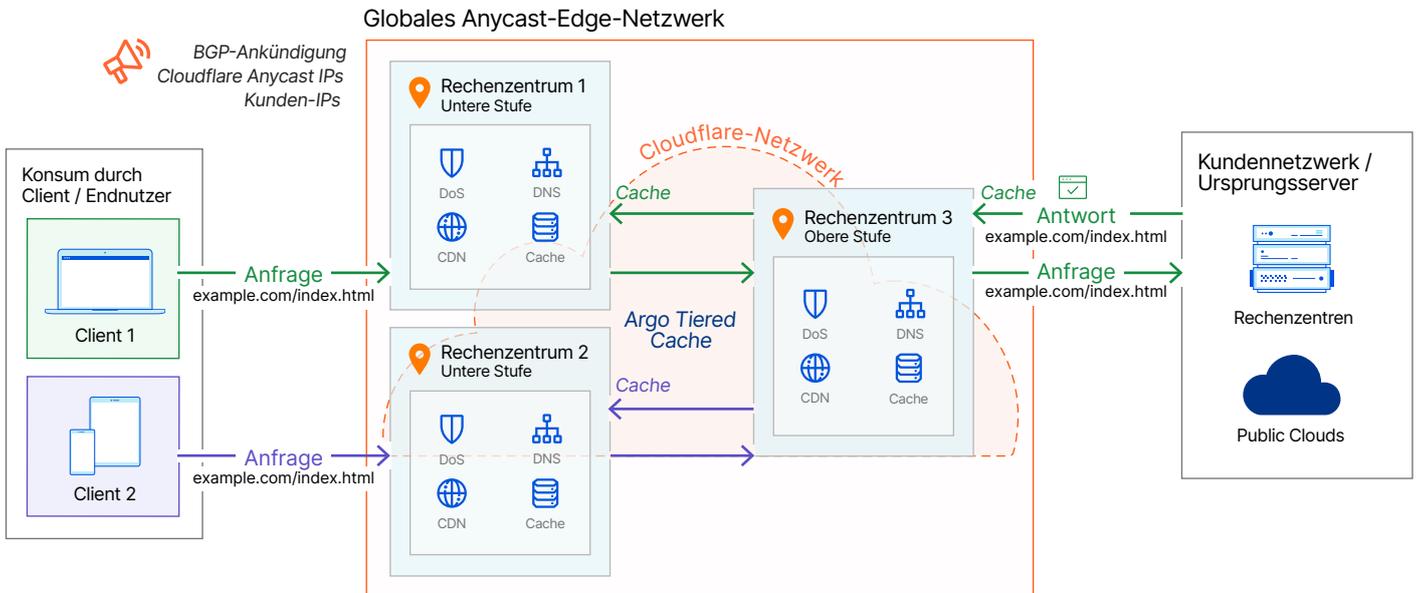


Abbildung 4: HTTP-Anfragen und Traffic-Fluss durch Cloudflare CDN

Client 1	Client 2
<ul style="list-style-type: none"> <li>Die erste Anfrage, die in Datenzentrum 1 eingeht, führt zu einem Cache-Fehler (Cache-Miss), da die Anfrage zuvor von keinem Client gestellt wurde.</li> <li>Es wurde kein zwischengespeicherter Inhalt gefunden, also wendet sich Rechenzentrum 1 an ein Rechenzentrum der oberen Stufe, um eine Kopie des Inhalts anzufordern.</li> <li>Das Rechenzentrum der oberen Stufe verfügt ebenfalls nicht über lokal zwischengespeicherte Inhalte, so dass es an den Ursprungsserver eine Anfrage für Inhalte stellt. Nach Erhalt der Inhalte speichert das Rechenzentrum der oberen Stufe diese lokal im Zwischenspeicher und leitet sie an das anfragende Rechenzentrum der unteren Stufe weiter. Das Rechenzentrum der unteren Stufe speichert die Inhalte im Cache und antwortet dem Client.</li> </ul>	<ul style="list-style-type: none"> <li>Die zweite Anfrage eines anderen Clients, die in Rechenzentrum 2 eingeht, führt zu einem Cache-Fehler, da die Anfrage zuvor von keinem Client in Rechenzentrum 2 gestellt wurde.</li> <li>Es wurde kein zwischengespeicherter Inhalt gefunden, also wendet sich Rechenzentrum 2 an ein Rechenzentrum der oberen Stufe, um eine Kopie des Inhalts anzufordern.</li> <li>Zwischengespeicherte Inhalte, die im Rechenzentrum der oberen Stufe gefunden wurden. Rechenzentrum 2 ruft diese Inhalte lokal ab, speichert sie zwischen und antwortet dem Client.</li> </ul>

## Traffic-Fluss: Argo Tiered Cache, Smart Tiered Cache-Topologie (Fortsetzung)

In Abbildung 4 zeigt der Traffic-Fluss von Client 1 den Traffic-Fluss, wenn eine Client-Anfrage bei einem Rechenzentrum eingeht, das dem Client am nächsten ist, nämlich Rechenzentrum 1. Da auf dem eingehenden Rechenzentrum nichts lokal zwischengespeichert ist und das Tiered Caching aktiviert ist, wird eine Anfrage an das Rechenzentrum der oberen Stufe gesendet, um eine Kopie des zwischenzuspeichernden Inhalts anzufordern.

Da das Rechenzentrum der oberen Stufe den Inhalt ebenfalls nicht zwischengespeichert hat, sendet es die Anfrage an den Ursprungsserver, speichert den empfangenen Inhalt bei der Antwort und antwortet dem Rechenzentrum der unteren Stufe mit dem zwischengespeicherten Inhalt. Das Rechenzentrum der unteren Stufe speichert die Inhalte im Cache und antwortet dem Client.

## Argo Smart Routing

Argo Smart Routing ist ein Dienst, der optimierte Routen durch das Cloudflare-Netzwerk findet, um Antworten noch schneller an die Benutzer zu liefern. Wie bereits erwähnt, nutzt Cloudflare CDN Argo Smart Routing, um die besten Rechenzentren der oberen Stufe für Argo Tiered Cache zu ermitteln.

Darüber hinaus kann Argo Smart Routing aktiviert werden, um sicherzustellen, dass jederzeit die schnellsten Wege über das Cloudflare-Netzwerk zwischen den Rechenzentren der oberen Stufe und den Ursprungsservern genommen werden. Ohne Argo Smart Routing wird die Kommunikation zwischen den Rechenzentren der oberen Stufe und den Ursprungsservern immer noch auf intelligente Weise um Probleme im Internet herum geleitet, um die Erreichbarkeit des Ursprungsservers zu gewährleisten.

Beachten Sie, dass bei einer neuen Anforderung desselben Inhalts an ein anderes Rechenzentrum (Client 2-Traffic-Fluss), nämlich Rechenzentrum 2, der Inhalt nicht lokal zwischengespeichert wird. Stattdessen wird der Inhalt vom Rechenzentrum der oberen Stufe abgerufen, wo er bei der ersten Anforderung desselben Inhalts zwischengespeichert wurde.

Da das Rechenzentrum der oberen Stufe den zwischengespeicherten Inhalt für die zweite Anfrage zurückgibt, wird der Weg zum Ursprungsserver vermieden. Dies führt zu höheren Cache-Trefferquoten, schnelleren Antwortzeiten, eingesparten Bandbreitenkosten zwischen dem Cloudflare-Netzwerk und dem Ursprungsserver und einer geringeren Belastung des Ursprungsservers bei der Beantwortung von Anfragen.

Argo Smart Routing beschleunigt den Traffic, indem es Echtzeitdaten und Netzwerkinformationen aus dem Routing von über 28 Millionen HTTP-Anfragen pro Sekunde berücksichtigt. Es stellt sicher, dass die schnellsten und zuverlässigsten Netzwerkpfade über das Cloudflare-Netzwerk zum Ursprungsserver geführt werden. Im Durchschnitt sorgt Argo Smart Routing für eine 30 % schnellere Performance bei Web-Ressourcen.

# CLOUDFLARE CDN-REFERENZARCHITEKTUR

## Traffic-Fluss: Argo Tiered Cache, Smart Tiered Cache-Topologie mit Argo Smart Routing

Abbildung 5 zeigt den Traffic-Fluss, wenn Argo Tiered Cache und Argo Smart Routing nicht aktiviert sind. Die Anfrage kommt im nächstgelegenen Rechenzentrum an, und da der Inhalt nicht lokal zwischengespeichert wird und Argo Tiered Cache nicht aktiviert ist, wird die Anfrage direkt an den Ursprungsserver für den Inhalt gesendet. Da Argo Smart Routing nicht aktiviert ist, wird bei der Kommunikation mit dem Ursprungsserver ein zuverlässiger, aber vielleicht nicht der schnellste Weg gewählt.

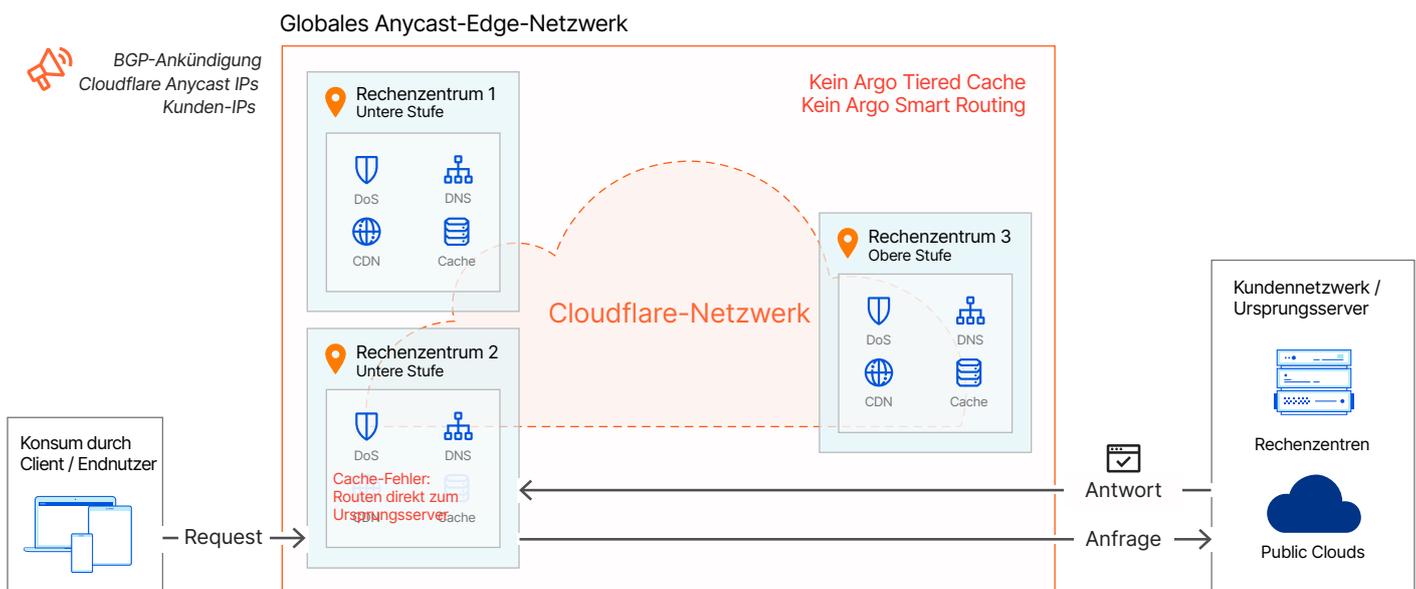


Abbildung 5: Cloudflare CDN ohne Argo Tiered Cache und Argo Smart Routing

# CLOUDFLARE CDN-REFERENZARCHITEKTUR

## Traffic-Fluss: Argo Tiered Cache, Smart Tiered Cache-Topologie mit Argo Smart Routing (Fortsetzung)

Abbildung 6 zeigt den Traffic-Fluss, wenn sowohl Argo Tiered Cache als auch Argo Smart Routing aktiviert sind.

In Abbildung 6 wird, wenn eine Anfrage von Rechenzentrum 1 empfangen wird und ein Cache-Fehler vorliegt, der Cache des Rechenzentrums der oberen Stufe, Rechenzentrum 3, überprüft. Wenn der zwischengespeicherte Inhalt im Rechenzentrum der oberen Stufe nicht gefunden wird, wird die Anfrage bei aktiviertem Argo Smart Routing auf dem schnellsten Weg vom Rechenzentrum der oberen Stufe zum Ursprungsserver gesendet.

Der schnellste Pfad wird durch die Fähigkeiten der Netzwerkinformationen von Argo bestimmt, die Echtzeit-Netzwerkdaten wie Überlastung, Latenz und RTT berücksichtigen.

Mit dem Cloudflare CDN wird Argo Smart Routing in folgenden Situationen verwendet:

1. Es liegt ein Cache-Fehler vor und die Anfrage muss an den Ursprungsserver gesendet werden, um den Inhalt abzurufen,
2. Es gibt eine Anfrage für nicht zwischenspeicherbare Inhalte, wie z. B. dynamische Inhalte (z. B. APIs), und die Anfrage muss an den Ursprungsserver gehen.

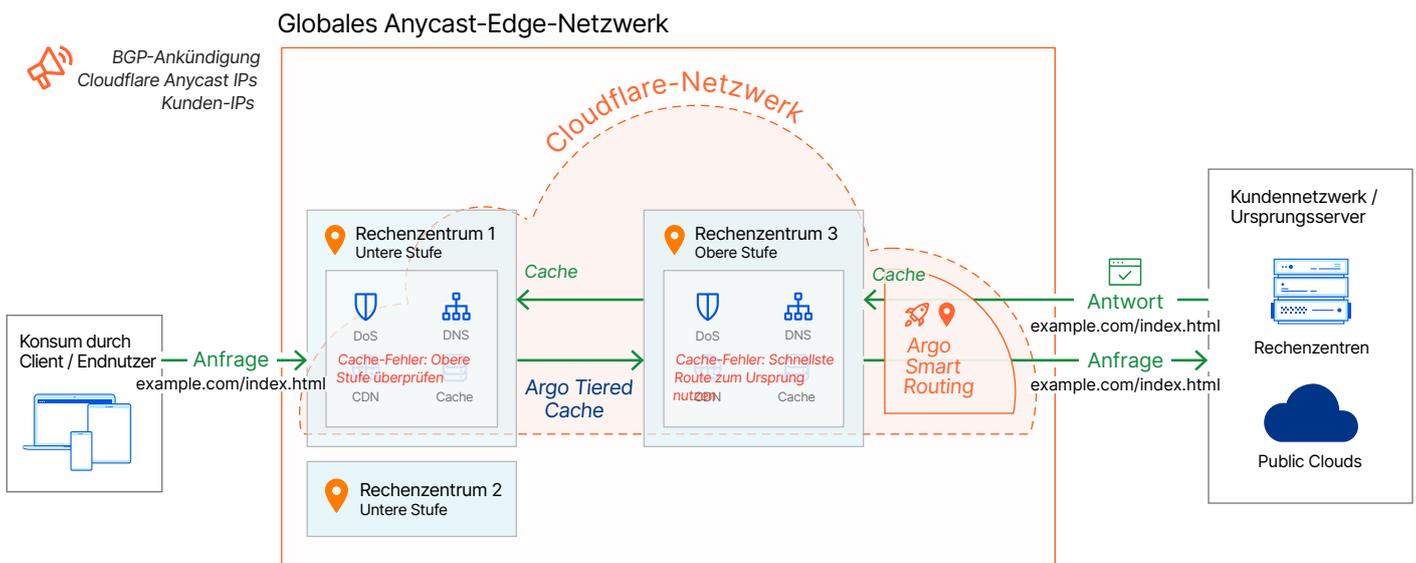


Abbildung 6: Cloudflare CDN mit Argo Tiered Cache und aktiviertem Argo Smart Routing

# Zusammenfassung

---

Zusammenfassend lässt sich sagen, dass das Cloudflare CDN eine SaaS-Lösung ist, die Kunden bei der Bewältigung der Herausforderungen in Bezug auf Latenz, Performance, Verfügbarkeit, Redundanz, Sicherheit und Kosten hilft. Das Cloudflare CDN nutzt das globale Anycast-Edge-Netzwerk von Cloudflare und den Argo Tiered Cache, um optimierte Ergebnisse zu liefern und gleichzeitig Kosten für Kunden zu sparen. Kunden können auch Argo Smart Routing aktivieren, um sicherzustellen, dass der schnellste Netzwerkpfad verwendet wird, um Anfragen an den Ursprungsserver weiterzuleiten.

---

© 2022 Cloudflare Inc. Alle Rechte vorbehalten. Das Cloudflare-Logo ist ein Markenzeichen von Cloudflare. Alle weiteren Unternehmens- und Produktnamen sind ggf. Markenzeichen der jeweiligen Unternehmen.