**IN THE UNITED STATES DISTRICT COURT**
**FOR THE EASTERN DISTRICT OF TEXAS**
**MARSHALL DIVISION**

| | |
|---|---|
| Sandpiper CDN, LLC,<br><br>*Plaintiff*,<br><br>v.<br><br>Microsoft Corporation,<br><br>*Defendant*. | Civil Case No. _____<br><br>**JURY TRIAL DEMANDED** |

## COMPLAINT FOR PATENT INFRINGEMENT

Plaintiff Sandpiper CDN, LLC ("Sandpiper CDN," "Sandpiper," or "Plaintiff") hereby files this Complaint for patent infringement against Defendant Microsoft Corporation ("Microsoft" or "Defendant"), and alleges as follows:

## NATURE OF THE ACTION

1. This is a civil action against Microsoft for patent infringement arising under the patent statutes of the United States, 35 U.S.C. § 271, *et seq.*, for the infringement of United States Patent Nos. 8,478,903, 8,924,466, 9,456,053, 9,722,883, 9,762,692, and 10,701,173 (collectively, "the Asserted Patents"). A true and correct copy of each Asserted Patent is attached to this Complaint as Exhibits [A-F], respectively. Each of the Asserted Patents is owned by Plaintiff Sandpiper CDN, and Plaintiff and/or its predecessors-in-interest have satisfied all statutory obligations required to collect pre- and post-filing damages for the full period allowed by law for infringement of the Asserted Patents, including compliance with 35 U.S.C. § 287.

## PARTIES

2. Plaintiff Sandpiper CDN is a Delaware limited liability company with its principal place of business in Wilmington, Delaware.

3. Defendant Microsoft Corporation is a Delaware corporation with a principal place of business at One Microsoft Way, Redmond, WA 98052. Upon information and belief, Microsoft has been registered to do business in Texas since March 13, 1995, and may be served with process via its registered agent: Corporation Service Company, d/b/a CSC – Lawyers Incorporating Service Company, 211 E. 7th Street, Suite 620, Austin, TX 78701.

## JURISDICTION AND VENUE

4. This action arises under the patent statutes of the United States, Title 35 of the United States Code. This Court has subject matter jurisdiction over this action under 28 U.S.C. §§ 1331 and 1338(a).

5. This Court has personal jurisdiction over Microsoft in this action because Microsoft conducts business in and has committed acts of patent infringement within this District and the State of Texas and has established minimum contacts with the forum such that the exercise of jurisdiction over Microsoft would not offend traditional notions of fair play and substantial justice.

6. Microsoft directly and/or through subsidiaries and intermediaries has engaged in continuous, systematic, and substantial activities within this State, including substantial marketing, offering, and sales of products and services. These products and services include Microsoft products and services comprising the accused Content Delivery Network ("CDN") functionalities.

7. Venue is proper in this District pursuant to 28 U.S.C. §§ 1391(b) and (c) and/or 1400(b). Defendant maintains regular and established places of business in the Eastern District of Texas and the State of Texas, regularly transacts business in the Eastern District of Texas, and has

committed and continues to commit acts of patent infringement in the Eastern District of Texas.

8. Personal jurisdiction over Defendant Microsoft is proper in this District because, among other things, Microsoft has seven corporate offices in the State of Texas and employs hundreds of people therein. Microsoft represents that one of those offices is in Frisco, Texas, and thus within this District.[1]

9. Additionally, Microsoft operates Microsoft Windows Stores within at least ten Best Buy retail locations throughout the State of Texas, including in this District. According to Microsoft, these Windows Stores are a "comprehensive store-within-a-store," with dedicated "Microsoft specialists" serving Best Buy customers within this "unique environment."[2] These locations are operated by Microsoft within Best Buy stores and are regular and established places of business for Microsoft. Indeed, as Microsoft's vice president of computing for Best Buy put it, "The Windows Store creates the kind of retail destination we all want to shop in, combining great selection, the latest technology, the best service and the lowest prices."[3] Microsoft maintains its distinct business identity at these locations, renting the space from Best Buy and holding out its Windows Stores as discrete retail locations.[4]

10. Further, Microsoft owns and maintains millions of dollars of real property in Collin County, located within this District.[5]

---

[1] Microsoft, *Microsoft U.S. Office Locations* (available at https://www.microsoft.com/en-us/about/officelocator/all-offices?msockid=325270f10c6b61543e3b64490de760b4) (last accessed June 9, 2025).

[2] Microsoft, *Microsoft and Best Buy announce the Windows Store only at Best Buy*, (available at https://news.microsoft.com/source/2013/06/13/microsoft-and-best-buy-announce-the-windows-store-only-at-best-buy-2/) (last accessed June 9, 2025).

[3] *Id.*

[4] Microsoft, *Talking Retail: The New Windows Store Only at Best Buy* (June 13, 2013), (available at https://blogs.windows.com/windowsexperience/2013/06/13/talking-retail-the-new-windows-store-only-at-best-buy/) (last accessed June 9, 2025).

[5] *See* Collin Central Appraisal District, *Property Search* (available at https://esearch.collincad.org) (last accessed June 9, 2025) (search results for "Microsoft").

11. Microsoft similarly owns and maintains tens of thousands of dollars in business personal property registered at several Best Buy locations in Denton County, also within this District.[6]

12. Microsoft also holds approximately $2 million worth of business personal property, including data servers, at Aligned Data Center, located at 2800 Summit Avenue, Plano, TX 75074, which is within this District.

13. Microsoft's website confirms that its "Azure Content Delivery Network" has a "point of presence (POP) location[]" in "Plano, TX, USA."[7]

14. Upon information and belief, Microsoft's Azure POP location operates out of the Aligned Data Center located in this District.

15. Compounding its significant connections to this District, in 2023 Microsoft announced a multi-billion-dollar deal with specialist cloud computing provider CoreWeave to use its datacenters for some of Microsoft's Azure AI workloads.[8] This deal will include Microsoft's use of CoreWeave's $1.6 billion datacenter in Plano, Texas, located in this District.[9] Microsoft is currently CoreWeave's biggest customer.[10]

16. Courts have repeatedly found jurisdiction over Microsoft to be proper in the Eastern

---

[6] *See* Denton Central Appraisal District, *Property Search* (available at https://www.dentoncad.com/property-search) (last accessed June 9, 2025) (search results for "Microsoft").

[7] Microsoft, "Azure Content Delivery Network Coverage by Metro," Sept. 27, 2024 (available at https://learn.microsoft.com/en-us/azure/cdn/cdn-pop-locations) (last accessed June 9, 2025).

[8] *See* Sebastian Moss, *Datacenter Dynamics*, "Microsoft signs multi-billion dollar deal with GPU cloud provider CoreWeave to meet AI needs" (available at https://www.datacenterdynamics.com/en/news/microsoft-signs-multi-billion-dollar-deal-with-gpu-cloud-provider-coreweave-to-meet-ai-needs/) (last accessed June 9, 2025).

[9] *See* Sebastian Moss, *Datacenter Dynamics*, "CoreWeave plans $1.6bn AI cloud data center in Plano, Texas" (available at https://www.datacenterdynamics.com/en/news/coreweave-plans-16bn-ai-cloud-data-center-in-plano-texas/) (last accessed June 9, 2025).

[10] *See* Reinhardt Krause, *CoreWeave Stock Rockets to New High Amid New Data Center Leasing Deal*, Investor's Business Daily, (available at https://www.investors.com/news/technology/coreweave-stock-new-high-nvidia-stock-applied-digital/) (last accessed June 9, 2025).

District of Texas, including in numerous patent disputes where Microsoft consented to jurisdiction for or litigated in this District.[11]

17. Microsoft has targeted and continues to direct its business activities toward this District. Given Microsoft's substantial business and property in this District and the State of Texas, and because of its ongoing infringing activities in this District, Microsoft is subject to this Court's general and specific jurisdiction pursuant to due process and the Texas Long Arm statute.

18. Venue is likewise proper in this District because of Microsoft's numerous physical places of business, employees, and property ownership in this District, and because Microsoft has repeatedly consented to litigation in this District. Moreover, as described more fully below, on information and belief Microsoft commits acts of infringement in this District, including by performing the methods of the Asserted Claims (or at least one or more steps of these method claims) of the Asserted Patents through its CDN POP(s) and/or data centers in this District and otherwise delivering content to end users in this District through the accused CDN services.

## SANDPIPER NETWORKS REVOLUTIONIZES CONTENT DELIVERY NETWORKS

19. Today, content delivery networks ("CDN") enable content providers to quickly deliver online information to millions of consumers simultaneously. But this has not always been the case.

20. In the early 1990s, the Internet rapidly evolved from burgeoning technology to household staple in the span of a few years. This mass adoption led to data congestion issues, as a quickly expanding user base sought simultaneous access to Internet content. The typical computer

[11] *See, e.g., Dialect, LLC v. Microsoft Corp.*, 2:24-cv-01067-JRG, Dkt. 13 at paras 12-16 (E.D. Tex. May 19, 2025) (admitting for purposes of this case that Microsoft is subject to personal jurisdiction in EDTX); *VPN Tech. Holdings, LLC v. Microsoft Corp.*, 2:25-cv-00001-JRG-RSP, Dkt. 12 at paras 5-7 (Feb. 24, 2025) (admitting personal jurisdiction and specific jurisdiction in EDTX for this case); *i4i Ltd. P'ship v. Microsoft Corp.*, 398 F. Supp. 3d 90 (E.D. Tex. 2019) (declining to contest personal jurisdiction in patent infringement litigation); *Biscotti Inc. v. Microsoft Corp.*, 302 F. Supp. 3d 797 (E.D. Tex. 2018) (declining to contest personal jurisdiction in patent infringement litigation).

server in the 1990s was only capable of handling a limited number of simultaneous connections before it became overloaded, leading to congested network segments, overburdened servers, and sluggish load times. The problem was even more pronounced for Internet users who lived far from the physical servers hosting their content, who experienced more lag and higher latency due to the geographic distance their signals traveled through physical internet cables.

21. Andrew Swart and David Farber were among the first individuals to develop services that allowed content providers to avoid the common congestion and performance issues that plagued Internet transmission in the mid-1990s. One of their solutions was to deploy CDN servers around the world that would more evenly distribute where data was processed. Mr. Swart and Mr. Farber developed infrastructure that replicated content requested from customers' origin servers to appropriate CDN servers and transparently rendezvoused the request to the CDN server best able to deliver that content. Their invention helped transform the early Internet, making it more efficient, responsive, and adaptable to users. Unsurprisingly, this service and its architecture were quickly imitated by many others in the industry, including Microsoft.

22. The CDN technology developed by Mr. Swart and Mr. Farber connected consumers to an edge server with available bandwidth that was geographically closer to them. This revolutionary approach provided numerous technical benefits. For example, distributing content across a network of servers alleviated data congestion issues, while connecting consumers to nearby edge servers—rather than distant origin servers—reduced latency. Mr. Swart and Mr. Farber developed and built systems and methods for propagating data from origin servers to edge servers (in one example, a process known as "caching") for storage and delivery to clients based on network demand/traffic.

23. In 1996, Mr. Swart and Mr. Farber founded Sandpiper Networks Inc. ("Sandpiper Networks") to further develop and commercialize their novel CDN concepts. By at least May 24, 1996, the Sandpiper Networks team developed infrastructure for delivering streaming resources, such

as audio and video, using Sandpiper's CDN.

24. Sandpiper Networks labored not only to build and implement its CDN, but also to protect its groundbreaking innovation through patent protection. Recognizing that its inventions could revolutionize content delivery worldwide, Sandpiper Networks filed numerous patent applications directed to its foundational CDN technology.

25. From at least May 1998 and through 1999, Sandpiper Networks partnered directly with content providers to cache and deliver their content to end consumers through their CDN. Sandpiper Networks' first paying customer, the *L.A. Times*, paid Sandpiper Networks to host the report of Independent Counsel Ken Starr on his investigation of President Bill Clinton ("the Starr Report") beginning on September 11, 1998.

26. Sandpiper Networks continued gaining widespread notoriety and won numerous awards for its CDN products and services during this time.[12]

27. In December 1999, Sandpiper merged with Digital Island, Inc. ("Digital Island") in a deal valued at $1.1B (approximately $2.1B today, adjusting for inflation) with the aim of creating a global computer network that would facilitate consumer e-commerce transactions.[13] Digital Island subsequently filed additional patent applications directed to CDN technology.

28. Following a series of acquisitions, on or about 2004, the assets of Digital Island and

[12] *See* Nick Wingfield, *The Wall Street Journal*, *"Sandpiper Networks' Footprint Takes Aim at Internet Traffic*" (June 17, 1999), https://www.wsj.com/articles/SB929571998724587925?st=DAy23s&reflink=desktopwebshare_permalink; Karen Kaplan, *Los Angeles Times*, "Sandpiper Networks' Footprint Wins Award" (May 31, 1999), https://www.latimes.com/archives/la-xpm-1999-may-31-fi-42763-story.html; CBR Staff Writer, *Tech Monitor*, "Sandpiper Adds RealSystem G2 to its Content Delivery Network" (Aug. 4, 1999), https://www.techmonitor.ai/technology/sandpiper_adds_realsystem_g2_to_its_content_delivery_network); Barbara Murphy, *Los Angeles Times*, "Sandpiper Networks Given Investor's Choice Award" (Oct. 5, 1999) https://www.latimes.com/archives/la-xpm-1999-oct-05-me-18779-story.html.

[13] Karen Kaplan, *L.A. Times*, "Sandpiper Networks to Merge with Digital Island in $1.1-Billion Deal" (Oct. 9, 1999) https://www.latimes.com/archives/la-xpm-1999-oct-26-fi-26295-story.html.

the Sandpiper Networks' patents were purchased by the data center company Savvis Inc. ("Savvis").

29. In January 2007, Savvis' CDN service assets, including the CDNs and patents of Digital Island and Sandpiper Networks, were acquired by Level 3 Communications ("Level 3").

30. Following the January 2007 acquisition of Sandpiper's CDN and patents, Level 3 continued innovating upon the technologies described in the Sandpiper Networks and Digital Island patents. Level 3 patented these innovations and eventually became one of the foremost CDN operators in the U.S.

31. Level 3 was acquired by CenturyLink on or about November 2017, and the combined company rebranded as Lumen Technologies, or simply "Lumen," in September 2020.[14] Lumen is headquartered in Monroe, Louisiana, which is approximately a two-hour drive from Marshall, Texas.

32. Sandpiper CDN holds all rights and title to a portfolio of more than 400 patents resulting from the many years of research and development, hundreds of millions of dollars in capital investment, and ingenuity of numerous engineers employed by Sandpiper CDN and its predecessors.

## MICROSOFT'S INFRINGEMENT OF THE ASSERTED PATENTS

33. In the early-to-mid 2000s, as demand for CDNs continued its precipitous rise, numerous other companies entered the CDN market to take advantage of the increased demand. These companies commercialized their own CDNs by incorporating the foundational CDN technology pioneered and patented by Sandpiper Networks.

34. The misappropriation of Sandpiper Networks' patents capitalized on, and undermined, the significant financial investment and years of research into CDN development and implementation undertaken by Sandpiper Networks and its successors.

35. Microsoft is one such company. Beginning in the mid-2000s and throughout the

[14] *See* Lumen, "Level 3 Financing completes sale of Sustainability-Linked Senior Notes," (Jan. 13, 2021), https://ir.lumen.com/news/news-details/2021/Level-3-Financing-completes-sale-of-Sustainability-Linked-Senior-Notes/default.aspx.

2010s, as widely used Microsoft products such as the Office suite, the cloud computing platform Azure, and XBOX gaming systems expanded their online operations, Microsoft started relying heavily on third-party CDNs to efficiently deliver its content to consumers.

36. In 2007, for example, 95% of all Microsoft's online content was delivered by third-party CDNs. Level 3 provided Microsoft with CDN content delivery capabilities using the technology underlying its patents, including one or more of the Asserted Patents at this time.

37. Microsoft's General Manager for the Edge Computing Network acknowledged this reliance on Level 3 and third-party CDNs for content delivery in his keynote address at the 2009 Content Delivery Summit.[15] By then, as that presentation highlighted, Microsoft was determined to move its content delivery in-house. Microsoft projected that by 2010, it would drop its reliance on Level 3 and other third-party CDNs to just 40% of content delivery.

38. On or about February 2010, Microsoft's cloud computing platform Azure launched the Azure CDN to its subscribers, using technology described and claimed by one or more of the Asserted Patents. At no point did Microsoft license this patented technology from Sandpiper CDN or its predecessors.

39. From the launch of Azure CDN, and until on or about May 2018, the Azure CDN continued to offer CDN delivery options from third-party providers. This approach permitted users to choose third parties to provide content delivery through the Azure CDN network.[16]

40. Nevertheless, Microsoft continued to rely on third-party CDNs, including Level 3, throughout the 2010s for most of its content delivery. For example, in 2014, as one analyst estimated,

[15] Jeff Cohen Keynote Address Slide Deck, Content Delivery Summit (May 11, 2009), https://conferences.infotoday.com/documents/83/CDNSummit09-Keynote-Microsoft.pdf.
[16] Microsoft Azure Blog, "Announcing Microsoft's Own Content Delivery Network" (May 7, 2018), https://azure.microsoft.com/en-us/blog/announcing-microsoft-s-own-cdn-network/?msockid=325270f10c6b61543e3b64490de760b4.

Microsoft still required third-party CDNs for approximately 75% of its content delivery, including increased reliance on Level 3 and the Asserted Patents.[17]

41. On or about May 2018, Microsoft announced its own CDN, available as an individual provider on the Azure network. This allowed users to select Microsoft among its third-party CDN provider partners through its Azure CDN.[18]

42. Microsoft's CDN network, however, infringed the Asserted Patents.

43. Since 2018, Microsoft has greatly expanded its content delivery networks, including with the launch of a public CDN and multiple tiers of Azure CDN services, using technology claimed in the Asserted Patents. Moreover, Microsoft has never licensed the Asserted Patents from either Sandpiper CDN or its predecessors.

## MICROSOFT LEARNS OF THE ASSERTED PATENTS AND WILLFULY INFRINGES THEM

44. During at least part of Microsoft's infringement of the Asserted Patents, Microsoft either knew of the Asserted Patents, or was willfully blind to them, and disregarded a substantial risk of infringement, making Microsoft's infringement egregious and willful.

45. Sandpiper Networks and its successors widely publicized the Asserted Patents, which became well known as fundamental patents in the CDN and related spaces.

46. For example, on or about October 1999, when Sandpiper Networks announced its merger with Digital Island, including merged ownership of the Sandpiper Networks' patents, shares of both companies soared. By the close of trade on the day of the announcement, Sandpiper Networks was valued at over $1 billion, while Digital Island was valued at over $2 billion. Leo Spiegel, then

[17] Dan Rayburn, "Microsoft Relying More on Third Party CSNs, Limelight Networks Getting More Business," *Streaming Media Blog* (Feb. 17, 2015), https://www.streamingmediablog.com/2015/02/microsoft-third-party-cdns.html.
[18] "Announcing Microsoft's own Content Delivery Network," *Microsoft Azure Blog* (May 7, 2018), Announcing Microsoft's own Content Delivery Network | Microsoft Azure Blog.

CEO of Sandpiper Networks who became President of the combined company, estimated that given the growing need for CDNs, the market opportunity for the merged companies could exceed $20 billion.[19]

47. In December 2006, Level 3 announced its acquisition of the Sandpiper Networks and its predecessors' patents from Savvis for $135 million.[20] Reports at the time said that there were multiple bidders for these assets.[21] Around the time of this acquisition, an executive from Level 3 explained that Level 3 "acquired the business primarily for its intellectual property and architecture."[22] Several other publications commented on the importance of the Sandpiper CDN intellectual property, which includes several of the Asserted Patents and/or pending applications, to this sale.[23] An executive from Microsoft, Microsoft's then-general manager of Global Foundation

[19] Tech Monitor, "Digital Island Merges with Sandpiper Networks" (Oct. 15, 1999), https://www.techmonitor.ai/hardware/digital_island_merges_with_sandpiper_networks.

[20] LUMEN, "SAVVIS Hones Strategic Focus with Sale of CDN Business Services to Level 3 Communications" (Dec. 26, 2006), https://news.lumen.com/SAVVIS-Hones-Strategic-Focus-with-Sale-of-CDN-Services-Business-to-Level-3-Communications.

[21] Gigaom, "Is Savvis CDN Business For Sale?," aom/is-savvis-cdn-business-for-sale/ (last accessed June 9, 2025); *see also* Data Center Knowledge, "Report: SAVVIS Shopping Its CDN Network," (Oct. 23, 2006), https://www.datacenterknowledge.com/networking/report-savvis-shopping-its-cdn-network).

[22] Rich Miller, "Level 3 Readies Launch of CDN Network," Data Center Knowledge (May 11, 2007), https://www.datacenterknowledge.com/networking/level-3-readies-launch-of-cdn-network.

[23] *See* Gigaom, "Level3 Buys Savvis CDN Business," https://om.co/gigaom/level3-buys-savvis-cdn-business/ (noting how "all the intellectual property that comes with this buy" could help Level 3 become a top competitor in the CDN market) (last accessed June 9, 2025); Rich Miller, "Level 3 Acquires Savvis CDN Network," Data Center Knowledge (Dec. 26, 2006), https://www.datacenterknowledge.com/networking/level-3-acquires-savvis-cdn-network ("The deal includes network assets, customer contracts, and intellectual property used in Savvis' CDN business."); *see also* Denise Pappalardo, "Level 3 Completes Acquisition of Savvis' CDN Business," NetworkWorld (Jan. 23, 2007), https://www.networkworld.com/article/838219/lan-wan-level-3-completes-acquisition-of-savvis-cdn-business.html ("In the cash deal Level 3 picked up network assets, customer contracts and "intellectual property.""); Thomas, "Level 3 Completes Acquisition of SAVVIS Content Delivery Network," https://news.thomasnet.com/companystory/level-3-completes-acquisition-of-savvis-content-delivery-network-505868) (last accessed June 9, 2025) ("Pursuant to the definitive agreement, dated December 23, 2006, Level 3 has paid $132.5 million in cash to acquire certain assets, including network elements, customer contracts, and intellectual property used in SAVVIS's CDN business.").

Services, also publicly commented on this acquisition at the time, stating it might benefit Microsoft. Specifically, that Microsoft executive stated: "As we grow our online services business, stability and control over our network infrastructure becomes increasingly important to deliver great experiences for our customers, partners and advertisers. We look forward to a continued relationship with Level 3 as they embark upon this next phase of their network evolution."[24]

48. Just a few months later, in July 2007, Microsoft spent $200 million to end a co-location agreement with Savvis and become the direct lessee for two data centers. The same Microsoft executive, Microsoft's then-general manager of Global Foundation Services, publicly commented on this business deal, stating: "The acquisition of these assets is an important part of our vision for a globally scaled data center infrastructure that will keep pace with user demand for innovative online services."[25]

49. On information and belief, around December 2006 Microsoft was one of, if not the, largest customers of Savvis's CDN services. At the time that Savvis sold its CDN business, several outlets reported that numerous large technology companies were interested in the purchase, including because of the CDN patents held by Savvis. Shortly after the purchase, Microsoft announced its continued commitment to Level 3 as a customer, who had instead acquired Savvis's CDN business along with its patents. However, shortly thereafter and despite this knowledge, Microsoft chose to invest funds to build out and operate its own CDN network without owning or obtaining a license to Level 3's patents including the Asserted Patents.

50. Over the years, Microsoft has hired several individuals who previously worked at

[24] Brian Prince, Channel Insider, "Level 3 Communications to Buy SAVVIS CDN Division" (Dec. 27, 2006), https://www.channelinsider.com/news-and-trends/level-3-communications-to-buy-savvis-cdn-division/.

[25] Rich Miller, "Microsoft Assumes Savvis Leases for $200M," Data Center Knowledge, https://www.datacenterknowledge.com/hyperscalers/microsoft-assumes-savvis-leases-for-200m (last accessed June 9, 2025).

one of Sandpiper CDN's predecessor companies at a time when that predecessor owned one or more of Sandpiper CDN's foundational CDN patents while operating a CDN. On information and belief, at least one such individual lives in Dallas, TX. Microsoft, having acquired through these hirings the knowledge of those individuals, either knew of the Asserted Patents and the high likelihood that Microsoft's CDN activities infringe them, or was willfully blind to the same.

51. Furthermore, Sandpiper has been involved in significant, widely public litigation regarding infringement of its patents, including two of the Asserted Patents against Microsoft. These lawsuits include:

- *Sandpiper CDN, LLC v. Google LLC*, Case No. 2:24-cv-03951-AB (C.D. Cal. May 10, 2024). In May 2024, Sandpiper sued Google in the Central District of California, asserting patents within the same portfolio as those asserted in this case, including the '903 patent asserted against Microsoft in this case.
- *Sandpiper CDN, LLC v. Comcast Cable Comm's, LLC*, Case No. 2:24-cv-00886-JRG (E.D. Tex. Nov. 1, 2024). In November 2024, Sandpiper sued Comcast in the Eastern District of Texas, asserting patents within the same portfolio as those asserted in this case, including the '903 and '692 patents asserted against Microsoft in this case.

These lawsuits have received significant press coverage, including through press releases and articles from RPX.[26] On information and belief, Microsoft is a member of RPX and receives notices and updates from RPX on patent litigations, including those discussing the aforementioned lawsuits from

[26] *See, e.g.*, Peter Hayes, "Google Hit with Patent Lawsuit Over Content Delivery Network," Bloomberg, https://news.bloomberglaw.com/litigation/google-hit-with-patent-lawsuit-over-content-delivery-network-5 (last accessed June 9, 2025); RPX, "It's Not Every Patent Complaint That Name Checks 'The Starr Report' and Big Bad Voodoo Daddy," https://insight.rpxcorp.com/news/80905-it-s-not-every-patent-complaint-that-name-checks-the-starr-report-and-big-bad-voodoo-daddy (last accessed June 9, 2025); RPX, "Down an Early Two Against Google, Sandpiper CDN Sues Comcast," https://insight.rpxcorp.com/news/83502-down-an-early-two-against-google-sandpiper-cdn-sues-comcast.

Sandpiper.[27] As such, Microsoft was either aware of these lawsuits or willfully blind to them, and has never approached Sandpiper about a license during the pendency of these lawsuits.

52. Based at least on the foregoing, Microsoft's infringement of the Asserted Patents has been, is, and continues to be willful.

## THE INVENTION OF THE ASSERTED PATENTS DRIVES MICROSOFT'S PROFITABILITY

53. It would be difficult to overstate the value to Microsoft of having a CDN network that practices the Asserted Patents.

54. Microsoft leverages the inventions of the Asserted Patents both for a CDN network for Microsoft's internal use, which saves Microsoft the cost of leasing a CDN network from others for Microsoft's services (e.g., Windows updates, its Xbox gaming ecosystem, Office 365 tools, Teams and Skype videoconferencing, Azure content delivery, delivery of apps, games, and software through the online Microsoft Store, video streaming like Linkedin Learning, Microsoft Edge, and Bing), and for external use, with Microsoft leasing its Azure CDN network to third parties in exchange for fees.

55. In September 2022 it was reported that the "Largest contributor to [Microsoft's] +$30.2 billion revenue growth was Azure (+13.5bn / +45% growth)."[28]

56. In January 2024, Microsoft revealed in its quarterly report that its revenue from "intelligent cloud" products and related services had increased 22%, or $4.4 billion, in the fourth quarter of 2023 alone, "driven by Azure and other cloud services."[29]

---

[27] *See, e.g., Xockets, Inc. v. NVIDIA Corp. et al*, Case No. 6:24-cv-00453-LS, Dkt. 1 at ¶ 201 (W.D. Tex. Sept. 5, 2024) ("RPX was founded in 2008 and has more than 450 members, including . . . Microsoft."); Dkt. 128 at n.3 (discussing an RPX-Microsoft Third Amendment to Membership and License Agreement).

[28] Kamil Franek, "Microsoft Revenue Breakdown by Product, Segment and Country" https://www.kamilfranek.com/microsoft-revenue-breakdown/ (last accessed June 9, 2025).

[29] Microsoft Quarterly Report for Quarter Ending Dec. 31, 2023, U.S. Sec. and Exchange Comm., https://www.sec.gov/ixviewer/ix.html?doc=/Archives/edgar/data/789019/000095017024008814/msft-20231231.htm (last accessed June 9, 2025).

57. In April 2024 Microsoft posted a press release titled, "Microsoft Cloud strength fuels third quarter results."[30]

58. In May 2024, it was reported that Microsoft's "Intelligent Cloud segment is the largest source of profit, as well as the fastest-growing [segment of Microsoft]."[31]

59. In September 2024, another source explained: "At the heart of Microsoft's growth strategy lies its Microsoft Cloud services division, spearheaded by Microsoft Azure. As someone who's been closely following the cloud computing market, I've been impressed by Azure's meteoric rise. In fact, cloud computing has become Microsoft's most crucial revenue driver, with Azure leading the charge."[32]

60. Also in September 2024, another source explained: "Microsoft's trajectory changed when Satya Nadella, formally of the Azure division, became CEO in 2014 after working his way up through the company over the course of 19 years to president of the cloud business. The stock is up 1,000% in the ten years since Nadella took the helm using his multi-decade cloud experience to steer a remarkable turnaround from a corporate reputation mired in fighting open-source communities and anti-trust issues. Since Nadella became CEO, the returns in Microsoft's stock have exceeded Amazon and is tied with Apple, as of this writing."[33]

61. Microsoft's use of its own internal CDN is substantial. For instance, one publication estimates that in 2021 Microsoft comprised 3.32% of all internet traffic, and that in 2022 that number

---

[30] Microsoft, "Microsoft Cloud Strength Fuels Third Quarter Results" (Apr. 25, 2024), https://news.microsoft.com/2024/04/25/microsoft-cloud-strength-fuels-third-quarter-results-3/.

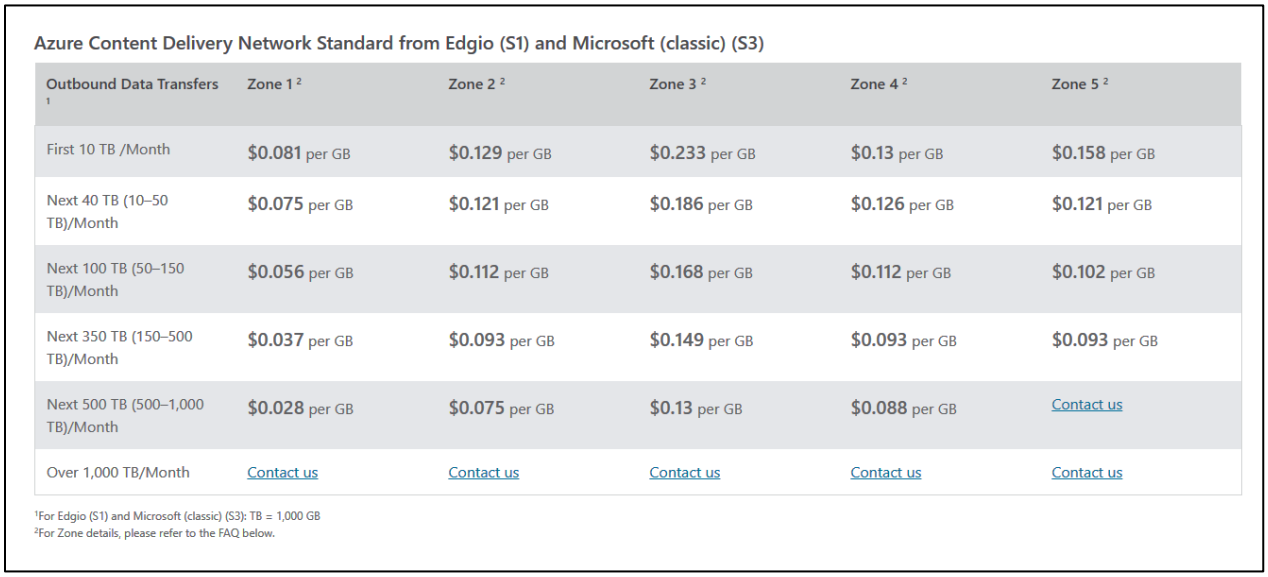
[31] Nathan Reiff, Investopedia, "How Microsoft Makes Money" (May 10, 2024), https://www.investopedia.com/how-microsoft-makes-money-4798809.

[32] Giro, "How Do Microsoft Make Money – A Comprehensive Breakdown," https://www.girolino.com/how-do-microsoft-make-money-a-comprehensive-breakdown/ (last accessed June 9, 2025).

[33] Beth Kindig, Forbes, "Prediction: Microsoft Azure to Reach $200 Billion in Revenue by 2028) (Sept. 5, 2024), https://www.forbes.com/sites/bethkindig/2024/09/05/prediction-microsoft-azure-to-reach-200-billion-in-revenue-by-2028/.

grew to 5.11%.[34]

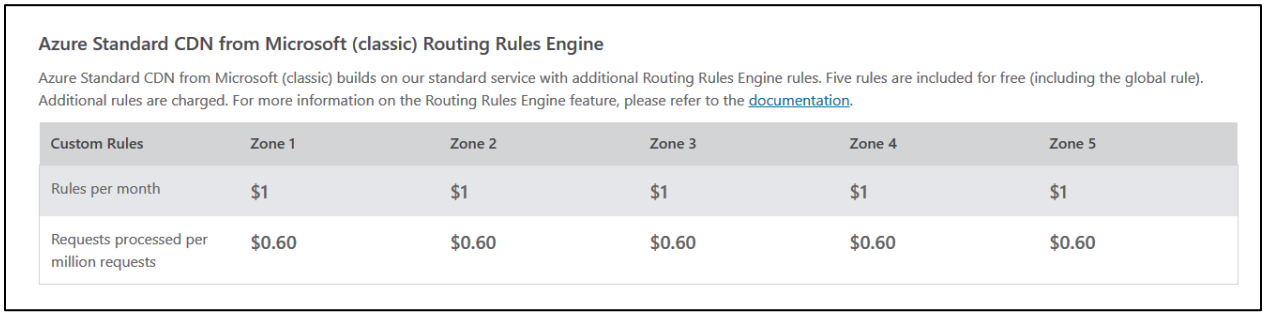
62. Microsoft posts the pricing for its external Azure CDN on its website, which is as follows:[35]



**Azure Content Delivery Network Standard from Edgio (S1) and Microsoft (classic) (S3)**

| Outbound Data Transfers [1] | Zone 1 [2] | Zone 2 [2] | Zone 3 [2] | Zone 4 [2] | Zone 5 [2] |
|---|---|---|---|---|---|
| First 10 TB /Month | $0.081 per GB | $0.129 per GB | $0.233 per GB | $0.13 per GB | $0.158 per GB |
| Next 40 TB (10–50 TB)/Month | $0.075 per GB | $0.121 per GB | $0.186 per GB | $0.126 per GB | $0.121 per GB |
| Next 100 TB (50–150 TB)/Month | $0.056 per GB | $0.112 per GB | $0.168 per GB | $0.112 per GB | $0.102 per GB |
| Next 350 TB (150–500 TB)/Month | $0.037 per GB | $0.093 per GB | $0.149 per GB | $0.093 per GB | $0.093 per GB |
| Next 500 TB (500–1,000 TB)/Month | $0.028 per GB | $0.075 per GB | $0.13 per GB | $0.088 per GB | Contact us |
| Over 1,000 TB/Month | Contact us | Contact us | Contact us | Contact us | Contact us |

[1]For Edgio (S1) and Microsoft (classic) (S3): TB = 1,000 GB
[2]For Zone details, please refer to the FAQ below.



**Azure Standard CDN from Microsoft (classic) Routing Rules Engine**

Azure Standard CDN from Microsoft (classic) builds on our standard service with additional Routing Rules Engine rules. Five rules are included for free (including the global rule). Additional rules are charged. For more information on the Routing Rules Engine feature, please refer to the documentation.

| Custom Rules | Zone 1 | Zone 2 | Zone 3 | Zone 4 | Zone 5 |
|---|---|---|---|---|---|
| Rules per month | $1 | $1 | $1 | $1 | $1 |
| Requests processed per million requests | $0.60 | $0.60 | $0.60 | $0.60 | $0.60 |

[34] Sandvine, Phenomena, The Global Internet Phenomena Report, at 10 (Jan. 2023), http://efaidnbmnnnibpcajpcglclefindmkaj/https://www.sandvine.com/hubfs/Sandvine_Redesign_2019/Downloads/2023/reports/Sandvine%20GIPR%202023.pdf.

[35] Microsoft, "Content Delivery Network Pricing," https://azure.microsoft.com/en-us/pricing/details/cdn/ (last accessed June 9, 2025).

Azure Content Delivery Network Premium from Edgio

| Outbound Data Transfers [1] | Zone 1 [2] | Zone 2 [2] | Zone 3 [2] | Zone 4 [2] | Zone 5 [2] |
|---|---|---|---|---|---|
| First 10 TB /Month | $0.158 per GB | $0.233 per GB | $0.466 per GB | $0.261 per GB | $0.317 per GB |
| Next 40 TB (10–50 TB)/Month | $0.14 per GB | $0.205 per GB | $0.396 per GB | $0.224 per GB | $0.27 per GB |
| Next 100 TB (50–150 TB)/Month | $0.121 per GB | $0.177 per GB | $0.335 per GB | $0.186 per GB | $0.228 per GB |
| Next 350 TB (150–500 TB)/Month | $0.102 per GB | $0.149 per GB | $0.279 per GB | $0.158 per GB | $0.196 per GB |
| Next 500 TB (500–1,000 TB)/Month | $0.093 per GB | $0.13 per GB | $0.242 per GB | $0.135 per GB | Contact us |
| Next 4,000 TB (1,000–5,000 TB)/Month | $0.084 per GB | $0.112 per GB | $0.21 per GB | $0.116 per GB | Contact us |
| Over 5,000 TB/Month | Contact us | Contact us | Contact us | Contact us | Contact us |

[1]For Edgio (S1) and Microsoft (classic) (S3): TB = 1,000 GB
[2]For Zone details, please refer to the FAQ below.

Acceleration data transfers from Edgio (S1)

| Outbound Data Transfers[1] | All Zones |
|---|---|
| First 50 TB /Month | $0.177 per GB |
| Next 100 TB (50–150 TB)/Month | $0.158 per GB |
| Next 350 TB (150–500 TB)/Month | $0.14 per GB |
| Next 500 TB (500–1,000 TB)/Month | $0.121 per GB |
| | Contact us |

[1]For Edgio (S1) and Microsoft (classic) (S3): TB = 1,000 GB

Azure CDN from Verizon has been rebranded to Azure CDN from Edgio.

Enterprise customers who would like to make minimum monthly traffic commitments are eligible to get discounted pricing. Please contact us to get more details.

63. As an indication of the scope of Microsoft's Azure CDN network, in connection with its Azure CDN services, Microsoft's web page for its external CDN services advertises that it "invests more than $1 billion annually on cybersecurity research and development" and "employ[s] more than 3,500 security experts who are dedicated to data security and privacy."[36]

## SANDPIPER CDN

64. Named after, and in homage to, the company that originally pioneered and developed

[36] Microsoft, "Azure Content Delivery Network," https://azure.microsoft.com/en-us/products/cdn/ (last accessed June 9, 2025).

CDN technologies in the 1990s, Sandpiper CDN now brings this suit to address Microsoft's infringement of the patented technology claimed by the Asserted Patents.

65. The Asserted Patents are valid and enforceable, and the inventions claimed in the Asserted Patents are enabled, novel, non-obvious, unconventional, and non-routine as of their respective filing dates.

## ASSERTED PATENTS

66. U.S. Patent No. 8,478,903 ("the '903 Patent") is entitled "Shared Content Delivery Infrastructure," and it claims priority to U.S. Patent Application No. 09/021,506, filed on February 10, 1998. *See* **Ex. A**.

67. As Internet use has increased, website owners must address ever-increasing bandwidth needs, dynamic changes in load, and performance issues relating to browsing clients, including clients in remote or distant locations.[37] When a server with website content, such as an origin server, receives multiple requests for website content, delivery of that content can be slow.

68. The inventions claimed by the '903 Patent provide concrete solutions to technical problems facing computer networks. In certain embodiments of the '903 Patent, methods are provided to, in a computer network, off-load processing of requests for selected resources (such as website content) by determining a different server to process those requests.[38] In some embodiments, this involves replicating content from a source associated with a client of a CDN network onto CDN servers. For example, end-user requests could be directed to the CDN servers instead of to the client's servers or the client's origin servers. These solutions in the '903 Patent improve the performance of providing website content to users in a network.[39]

69. Additionally, certain claimed embodiments of the '903 Patent solve issues related to

[37] '903 Patent, 1:27-31.
[38] *Id*. at 2:62-65.
[39] *Id.*

delivering resources from more than one content provider. For instance, the inventors of the '903 Patent pioneered CDN technology for delivering resources associated with more than one content provider, embodiments of which involve a shared CDN server and alias names in order to provide resources in response to requests. In one example, an alias name is an alternate name that can be used to make a connection. Certain claimed solutions of the '903 Patent include specific combinations that were not conventional at the time of the invention of the '903 Patent.

70. In one specific example of a technical improvement, a repeater server in a CDN maintains a partial mirror of more than one origin server by implementing a distributed and/or coherent cache of the origin server(s), in order to off-load processing and provide additional bandwidth for multiple website owners.[40] In one example, a repeater server is a network node that amplifies and rebroadcasts signals. In one example, a distributed cache is a type of cache that distributes data across multiple servers in a cluster, while a coherent cache ensures that all clients see the same data in a cache.

71. According to one claimed solution in the '903 Patent, a content delivery system with at least one shared repeater server is provided. The shared repeater server can replicate resources associated with origin servers. In one embodiment, requests for a resource on an origin server are directed, at least in part, based on an alias name to at least one repeater server.[41]

72. U.S. Patent No. 8,924,466 (the '466 Patent) is entitled "Server handoff in content delivery network" and it claims priority to U.S. Patent Application No. 11/715,316 filed March 8, 2007, which is a continuation of U.S. Patent Application No. 10/073,938 filed February 14, 2002. *See* **Ex. B**.

73. The inventions claimed by the '466 Patent provide concrete solutions to technical

[40] *Id*. at 4:37-40.
[41] *Id*. at claim 28.

problems facing computer networks. Certain of the inventions claimed by the '466 Patent generally relate to delivery of requested objects in a CDN by servers.[42] Objects are hosted at one or more parent servers and edge servers to replicate content such as audio, video, and image files.[43] Since it's typically not feasible to store all objects at edge servers due to their size and the limitations of storage space, cache servers are populated strategically using intelligent replication in order to optimize availability of popular content.[44] Additionally, to account for varying network congestion, redirection is used to route a client's request to a server that can provide the requested object either directly or indirectly.[45]

74. An exemplary embodiment involves a computer-implemented method for distributing objects maintained on at least one origin server via a CDN including providing a plurality of edge servers and parent servers distinct from the origin server.[46] A client's request is directed to a first edge server based at least in part on network traffic conditions and regardless of whether the first edge server has the requested object.[47] If the first edge server has the requested object, it is served to the client. Otherwise, the client is redirected to a second server such as a parent server to handle the request. If the requested object is not stored on the second server, the client is directed to another server in the CDN.[48]

75. U.S. Patent No. 9,456,053 (the '053 Patent) is entitled "Content delivery network" and it claims priority to U.S. Provisional Patent Application Nos. 61/570,448 and 61/570,486, each filed on December 14, 2011. *See* **Ex. C**.

---

[42] '466 Patent at Abstract.
[43] '466 Patent at 1:52-3:30.
[44] *Id.*
[45] *Id.*
[46] '466 Patent at Claim 1.
[47] *Id.*
[48] *Id.*

76. The ’053 Patent addresses challenges in maintaining updated per-customer configuration data as capacity-constrained networks are scaled. For example, when a new cache server is added to the CDN, that new cache needs to efficiently obtain customer configuration data so that it may be seamlessly integrated into the existing CDN network.[49]

77. The inventions claimed by the ’053 Patent provide concrete solutions to these technical problems by, for example, distributing configuration data to CDN cache servers from a control core. A control core operates as a geographically distributed cluster of machines connected via high-speed communication links acting as a database and web server combination.[50] The control core stores configuration data and controls CDN operations to ensure high availability, efficient operation, and seamless integration of internal CDN components such as cache servers.[51]

78. An exemplary embodiment of the ’053 Patent comprises a computer-implemented method. The method involves cache servers obtaining global configuration data from a control core, selectively updating this data as needed, and retrieving customer-specific configuration information.[52] Using this combined data, the cache servers deliver client-requested content tailored to the customer's requirements and the global configuration data.[53]

79. U.S. Patent No. 9,722,883 (the ’883 Patent) is entitled “Responsibility-based peering” and it claims priority to U.S. Provisional Patent Application No. 61/737,072 filed on December 13, 2012. *See* **Ex. D**.

80. One problem facing content delivery networks involved content libraries so voluminous that they would not fit on a single machine or cache server. To address this problem,

[49] ’053 Patent at Abstract.
[50] *Id.* at 4:43-5:30.
[51] *Id.*
[52] *Id.* at Claim 27.
[53] *Id.*

nodes could be configured in a "peer group" to assume one or more discrete responsibilities involved in collaborative processing of a request across the peer group.[54] Each node in the peer group may be assigned a set of capabilities to determine the responsibilities it may have, and the responsible nodes would be assigned from the set possessing these capabilities.[55] Ideally, responsibilities should change in a predictable way in the face of capability losses due to node failures, but as the '883 Patent notes, there is a tradeoff between the goals of consistency and load balancing.[56]

81. The '883 patent provides concrete solutions to certain of these technical problems by providing configurable policies defining a mapping of network resources to a request peering type. Each node in a peer group is evaluated for its capabilities and a capacity metric which may change dynamically based on events or instructions from a control network.[57] A peering policy defines responsibility chains for the peer group and maps resource types to request types. This ensures that different request types are processed by tailored responsibility chains, with responsible nodes selected based on their current capacities and capabilities with respect to a given request. The peering policy also uses a responsibility function to map requests to suitable nodes to balance dynamic load demands.[58]

82. Embodiments of the '883 Patent further utilize consistent hashing to minimize disruption in node assignments.[59] This responsibility-based peering arrangement aims to have responsibilities change in a predictable way to promote consistency and load balancing.[60] This helps to ensure that responsibility allocations remain robust even during transient conditions, such as node

[54] '883 Patent at 51:46-52:30.
[55] *Id.*
[56] *Id.*
[57] *Id.*
[58] *Id.*
[59] *Id.* at 52:24-53:25.
[60] *Id.*

failures or capacity fluctuations.[61]

83. U.S. Patent No. 9,762,692 ("the '692 Patent") is entitled "Handling long-tail content in a content delivery network (CDN)." The '692 Patent claims priority to U.S. Provisional Patent Application No. 61/042,412, filed on April 4, 2008. *See* **Ex. E**.

84. CDNs generally provide multiple servers to serve content. Issues can arise in CDNs when responding to requests for content, such as technical issues relating to handling requests and timely providing content. For example, certain content can become popular or fade into obscurity dynamically over time.[62] CDNs face technical issues with respect to serving content that is popular, or not popular, in a network with multiple servers.[63] Moreover, CDNs can experience increased latency and degraded service if content is cached at a particular server even if the content is not popular.[64] The risk of "thrashing" to serve content to all customers is also an issue for CDNs when content is unnecessarily cached.[65]

85. The inventors of the '692 Patent developed concrete, technical solutions to these problems, which are described in the '692 Patent. The '692 Patent describes infrastructure and logic for delivering content in a content delivery network including a first tier of servers.[66] In some embodiments, a server in the first tier of servers obtains a request for a resource that is available as part of a content provider's library. If the resource is not available at the server or a peer of the server, certain embodiments determine if the resource is popular. If the resource is determined to be popular, the server can obtain and serve the resource. If the resource is not determined to be popular, the requester can be directed to a second server in a second tier of servers, which serves the resource, for

[61] *Id.*
[62] '692 Patent at 2:46-51.
[63] *Id.*
[64] *Id.* at 10:1-22.
[65] *Id.*
[66] *Id.* at Claim 1.

example.[67] In some embodiments, a portion of the content provider's library comprising the second server is distinct from a portion on another server in the second tier.

86. The '692 Patent describes a CDN including a tiered server system. Certain embodiments claimed by the '692 Patent address the reality of storage limitations by setting forth a framework that obtains only popular content at a first tier of servers, thus both speeding up content delivery and preserving memory space for popular content in a first tier. The inventors of the '692 Patent pioneered concrete, technical solutions for using tiers of servers and specific processes for serving resources in a memory conscious manner that addresses latency issues. Certain claimed embodiments of the '692 Patent include particular implementations and combinations that were not conventional at the time of the invention of the '692 Patent, providing specific technical improvements to CDNs.

87. U.S. Patent No. 10,701,173 (the '173 Patent) is entitled "Caching in a content delivery framework" and it claims priority to U.S. Provisional Patent Application No. 14/570,512 filed on December 15, 2014. *See* **Ex. F**.

88. Certain of the inventions claimed by the '173 Patent generally relate to CDN caching based on policies. Caching in a CDN involves storing content to facilitate faster access and delivery. Cached content typically adheres to caching policies, which specify expiration rules and directives.[68] Policies may be overridden by subscribers through APIs or management portals to customize caching behavior.[69] For instance, a cache policy may specify a unique expiration time for a particular content type.

89. To avoid the costly process of invalidating cached data when policies change, CDNs

[67] *Id.*
[68] '173 Patent at 3:1-4:27.
[69] *Id.*

use "late-binding" to determine a caching policy dynamically during request processing, ensuring the most up-to-date rules are applied without forcing content invalidation.[70] When handling requests, CDNs access databases (e.g., RuleBases) to retrieve relevant caching policies and enforce them via cache nodes. These nodes decide whether cached content is still valid based on the policy's constraints and the current state of the cache. If no cached content is available or if it is stale, a refresh or reload is triggered as dictated by the policy. Internal and external policies can differ, enabling CDNs to optimize caching for various client needs.[71] For instance, internal policies may specify shorter expiration times to balance storage and performance, while external policies provided to clients ensure adherence to content expiration rules. This framework allows CDNs to manage content delivery efficiently, handle policy overrides flexibly, and adapt dynamically to client or subscriber needs while minimizing disruptions.[72]

90. An exemplary method of the '173 Patent provides a concrete technical solution to optimizing the efficient updating of caching policies. The exemplary method operates within a CDN to efficiently handle content requests.[73] Upon receiving a request, a CDN node checks if the content is cached locally and evaluates its acceptability for serving based on the associated cache policy, which includes expiration criteria.[74] If the cached content is valid, the node serves it; otherwise, it retrieves and serves a fresh version, ensuring up-to-date content delivery while optimizing performance.[75]

## THE ACCUSED MICROSOFT FUNCTIONALITIES

91. Microsoft's Azure is a cloud computing platform offering a range of cloud services

[70] *Id.* at 4:28-6:36.
[71] *Id.*
[72] *Id.*
[73] *Id.* at Claim 1.
[74] *Id.*
[75] *Id.*

for businesses and developers.[76] Core service categories include cloud computing, cloud storage, databases, networking, application hosting and development, and security. Microsoft additionally offers content delivery infrastructure and services through its Azure CDN products, as well as leveraging these products to deliver its own internal content, updates, and other services for Microsoft products. These Azure CDN products include Azure CDN Standard from Microsoft, Azure CDN Standard from Edgio, Azure CDN Premium from Edgio, and Azure Front Door Standard and Premium.[77]

Azure Content Delivery Network includes three products:

- Azure CDN Standard from Microsoft
- Azure CDN Standard from Edgio (formerly Verizon)
- Azure CDN Premium from Edgio (formerly Verizon).

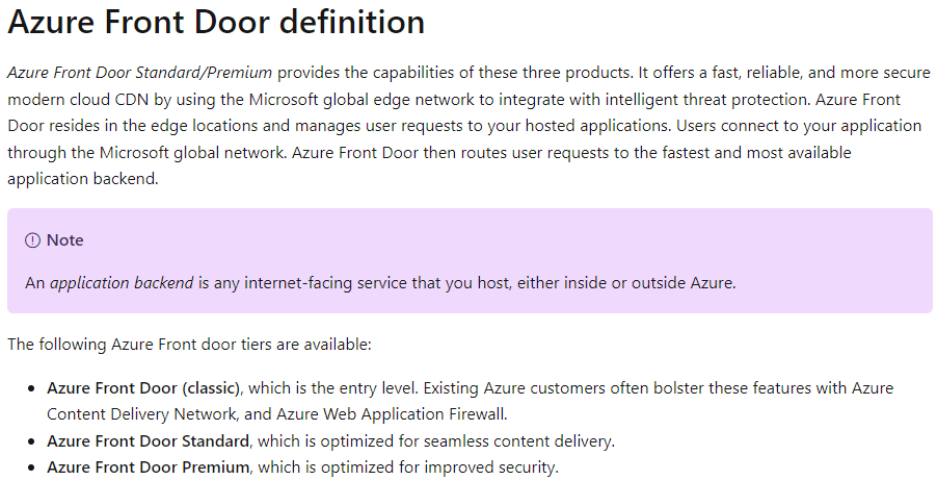
https://learn.microsoft.com/en-us/azure/cdn/cdn-features

92. Microsoft's Azure CDN products offer global coverage through distributed points of presence (PoPs) including throughout North America.[78] Microsoft's Azure CDN products utilize repeater servers (e.g., EdgeNodes or Endpoints) for replicating (e.g., caching) content from origin servers.[79] Azure CDN products include various performance features and optimizations including acceleration, delivery optimization, video streaming optimization, asset pre-loading, caching rules, customizable rules for content delivery, URL redirect/rewrite, custom DNS support, DDoS protection, authentication, and analytics.[80] Azure CDN was announced May 7, 2018 as a public preview.[81]

---

[76] https://azure.microsoft.com/en-us/resources/cloud-computing-dictionary/what-is-azure/
[77] https://learn.microsoft.com/en-us/azure/cdn/cdn-overview; https://learn.microsoft.com/en-us/azure/frontdoor/front-door-overview.
[78] https://azure.microsoft.com/en-us/products/cdn; https://learn.microsoft.com/en-us/azure/cdn/cdn-pop-locations.
[79] https://learn.microsoft.com/en-us/azure/cdn/cdn-overview
[80] https://learn.microsoft.com/en-us/azure/cdn/cdn-features
[81] https://azure.microsoft.com/en-us/blog/announcing-microsoft-s-own-cdn-network/

93. Azure Front Door is Microsoft's "modern" CDN product featuring similar content replication and delivery functionality with added scalability and performance.[82] Azure Front Door provides autoscaling and global traffic distribution with load balancing to ensure high availability.[83]



## Azure Front Door definition

*Azure Front Door Standard/Premium* provides the capabilities of these three products. It offers a fast, reliable, and more secure modern cloud CDN by using the Microsoft global edge network to integrate with intelligent threat protection. Azure Front Door resides in the edge locations and manages user requests to your hosted applications. Users connect to your application through the Microsoft global network. Azure Front Door then routes user requests to the fastest and most available application backend.

ⓘ Note

An *application backend* is any internet-facing service that you host, either inside or outside Azure.

The following Azure Front door tiers are available:

- **Azure Front Door (classic)**, which is the entry level. Existing Azure customers often bolster these features with Azure Content Delivery Network, and Azure Web Application Firewall.
- **Azure Front Door Standard**, which is optimized for seamless content delivery.
- **Azure Front Door Premium**, which is optimized for improved security.

https://learn.microsoft.com/en-us/training/modules/intro-to-azure-front-door/2-what-is-azure-front-door

94. Microsoft additionally offers Azure Service Fabric for building and managing scalable microservices and containerized applications. Azure Service Fabric provides virtual machine servers running clusters to collaboratively process requests related to microservices. Azure Service Fabric was made generally available March 31, 2016.[84]

95. Microsoft further offers Azure Arc, a cloud management platform that extends Azure capabilities to hybrid, multi-cloud, and edge environments for managing resources across diverse infrastructures from a single control plane.[85] For instance, Azure Arc allows management of resources including servers, Kubernetes clusters, Azure data services, SQL servers, and virtual machines.[86] Azure Arc was announced November 4, 2019.[87]

---

[82] https://learn.microsoft.com/en-us/azure/frontdoor/front-door-overview
[83] https://learn.microsoft.com/en-us/azure/frontdoor/front-door-faq
[84] https://azure.microsoft.com/en-us/blog/azure-service-fabric-is-ga/
[85] https://azure.microsoft.com/en-us/products/azure-arc/
[86] https://learn.microsoft.com/en-us/azure/azure-arc/overview
[87] https://azure.microsoft.com/en-us/blog/azure-services-now-run-anywhere-with-new-hybrid-capabilities-announcing-azure-arc/

96. Microsoft further offers Azure Policy, enabling organizations to create, assign, and manage policies to enforce compliance and security across their cloud resources.[88] Core features of Azure Policy include resource consistency ensuring uniform configuration across environments as well as security enhancement enforcing restrictions on resource configurations and services.[89]

97. Microsoft further offers Azure Resource Manager, providing infrastructure management and access control functionality in a unified management platform.[90] Azure Resource Manager was introduced in 2014.[91]

98. On information and belief, Microsoft operates its CDNs for both Microsoft's internal use (e.g., Windows updates, its Xbox gaming ecosystem, Office 365 tools, Teams and Skype videoconferencing, Azure content delivery, delivery of apps, games, and software through the online Microsoft Store, video streaming like Linkedin Learning, Microsoft Edge, and Bing), and for external use, with Microsoft leasing its Azure CDN network to third parties.

99. The aforementioned products and services are collectively the "Accused Microsoft Functionalities."

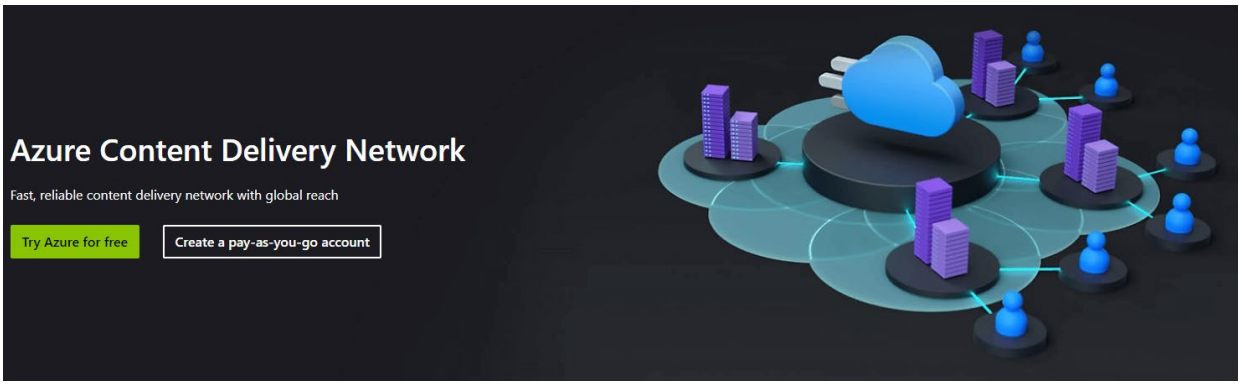
## COUNT I: INFRINGEMENT OF THE '903 PATENT

100. Plaintiff hereby incorporates by reference each of the allegations in the foregoing paragraphs as though fully set forth herein, and further alleges as follows.

101. Microsoft infringes at least claim 28 of the '903 Patent through one or more of the Accused Microsoft Functionalities. For instance, Microsoft Azure CDN comprises a plurality of origin servers hosting associated resources as well as shared repeater servers (e.g., EdgeNodes or Endpoints) to replicate resources associated with the origin servers.

---

[88] https://learn.microsoft.com/en-us/azure/governance/policy/overview
[89] https://sonraisecurity.com/blog/what-is-azure-policy-all-you-need-to-know/
[90] https://learn.microsoft.com/en-us/azure/azure-resource-manager/management/overview
[91] https://web.archive.org/web/20161029053752/https://azure.microsoft.com/en-us/documentation/articles/resource-manager-deployment-model/




https://azure.microsoft.com/en-us/products/cdn

**Increase speed and scalability for an optimal user experience**

Azure Content Delivery Network offers a global solution for rapidly delivering content. Save bandwidth and improve responsiveness when encoding or distributing gaming software, firmware updates, and IoT endpoints. Reduce load times for websites, mobile apps, and streaming media to increase user satisfaction globally.

https://azure.microsoft.com/en-us/products/cdn

Stream media and download large files quickly with optimized delivery.

Provide a scalable solution to handle worldwide traffic spikes and instantaneous high loads.

Integrate seamlessly with your Azure services to activate within minutes.

Protect content with custom domain HTTPS, DDoS, and WAF protection.

https://azure.microsoft.com/en-us/products/cdn

## Improved performance and user experience

Users expect a fast, reliable, and personalized web experience wherever they are. Content Delivery Network helps reduce latency and improve performance for high-bandwidth content by distributing user requests and serving content directly from edge servers. This brings the content closer to users and sends less traffic to the origin point, delivering superior online experiences.

https://azure.microsoft.com/en-us/products/cdn

## Global coverage and massive scalability

Scale on the fly with no downtime using a platform that responds to traffic fluctuations. A distributed, global presence enables Content Delivery Network to handle sudden traffic spikes and heavy loads—for example, during major product launches or global sporting events—without new infrastructure costs or capacity concerns. Global, distributed points of presence (PoPs) ensure fast delivery of content and protects the origin from excessive load.

https://azure.microsoft.com/en-us/products/cdn

102. Microsoft associates the at least one repeater server with a first alias name, wherein requests for a first resource located on a first origin server are directed, based at least in part on said first alias name, to the at least one repeater server for delivery of the first resource. For instance, Microsoft provides instructions on creating alias records associated with Azure CDN endpoints for users of both Azure DNS and third-party DNS providers.

# Tutorial: Add a custom domain to your endpoint

Article • 03/21/2024 • 22 contributors

Feedback

In this article

- Prerequisites
- Create a CNAME DNS record
- Add a custom domain to your content delivery network endpoint
- Verify the custom domain

Show 2 more

This tutorial shows how to add a custom domain to an Azure Content Delivery Network endpoint.

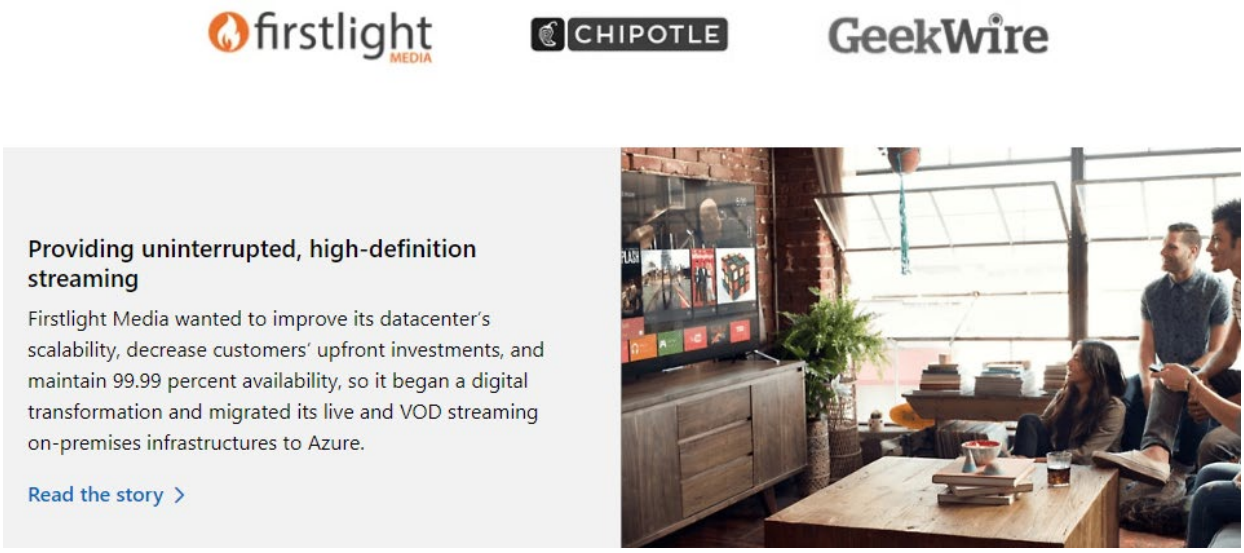
The endpoint name in your content delivery network profile is a subdomain of azureedge.net. By default when delivering content, the content delivery network profile domain gets included in the URL.

For example, `https://contoso.azureedge.net/photo.png`.

Azure Content Delivery Network provides the option of associating a custom domain with a content delivery network endpoint. This option delivers content with a custom domain in your URL instead of the default domain.

https://learn.microsoft.com/en-us/azure/cdn/cdn-map-content-to-custom-domain

103. Further, Microsoft Azure CDN supports custom domain alias creation for multiple CDN hosting clients accessing content from various origin servers cached by Azure CDN endpoints.




https://azure.microsoft.com/en-us/products/cdn#customer-stories

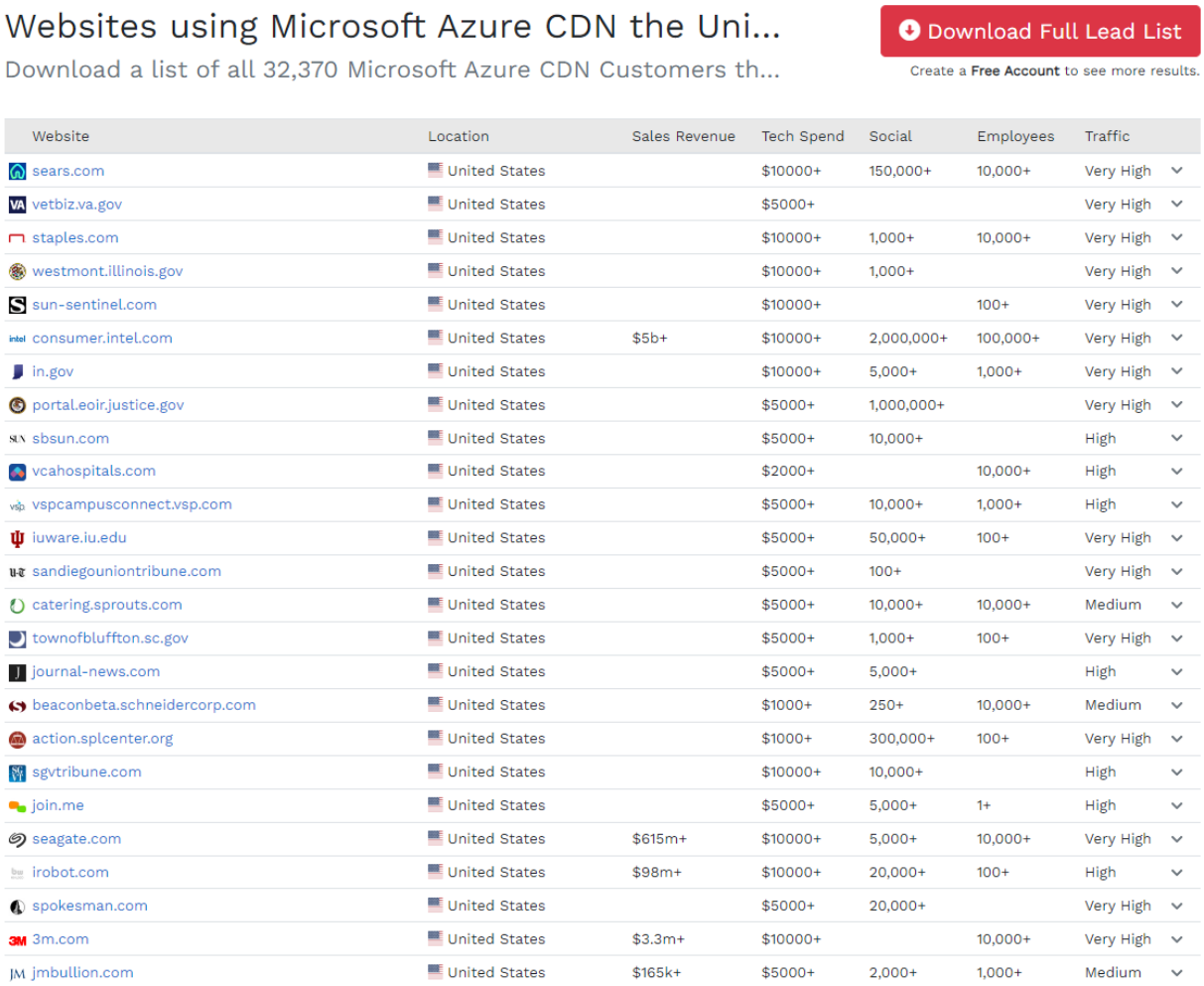
## Verify the custom domain

After you've completed the registration of your custom domain, verify that the custom domain references your content delivery network endpoint.

1. Ensure that you have public content that you want cached at the endpoint. For example, if your content delivery network endpoint is associated with a storage account, Azure Content Delivery Network caches the content in a public container. Set your container to allow public access and it contains at least one file to test the custom domain.

2. In your browser, navigate to the address of the file by using the custom domain. For example, if your custom domain is `www.contoso.com`, the URL to the cached file should be similar to the following URL: `http://www.contoso.com/my-public-container/my-file.jpg`. Verify that the result is that same as when you access the content delivery network endpoint directly at <**endpoint-hostname**>.azureedge.net.

https://learn.microsoft.com/en-us/azure/cdn/cdn-map-content-to-custom-domain



Websites using Microsoft Azure CDN the Uni...

Download a list of all 32,370 Microsoft Azure CDN Customers th...

Download Full Lead List

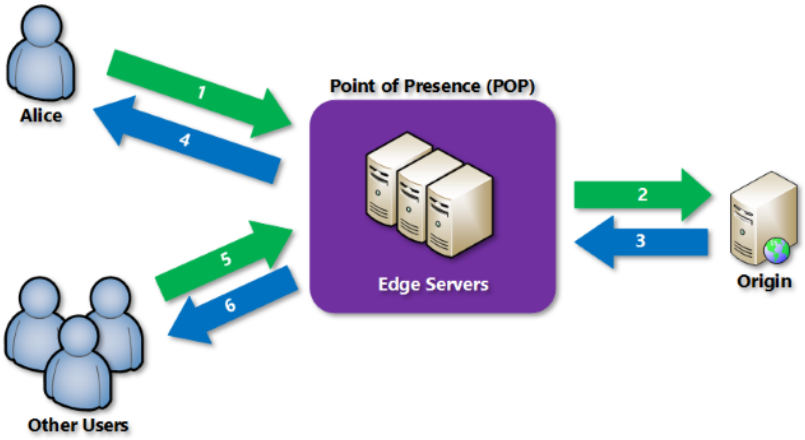
Create a **Free Account** to see more results.

| Website | Location | Sales Revenue | Tech Spend | Social | Employees | Traffic |
|---|---|---|---|---|---|---|
| sears.com | United States | | $10000+ | 150,000+ | 10,000+ | Very High |
| vetbiz.va.gov | United States | | $5000+ | | | Very High |
| staples.com | United States | | $10000+ | 1,000+ | 10,000+ | Very High |
| westmont.illinois.gov | United States | | $10000+ | 1,000+ | | Very High |
| sun-sentinel.com | United States | | $10000+ | | 100+ | Very High |
| consumer.intel.com | United States | $5b+ | $10000+ | 2,000,000+ | 100,000+ | Very High |
| in.gov | United States | | $10000+ | 5,000+ | 1,000+ | Very High |
| portal.eoir.justice.gov | United States | | $5000+ | 1,000,000+ | | Very High |
| sbsun.com | United States | | $5000+ | 10,000+ | | High |
| vcahospitals.com | United States | | $2000+ | | 10,000+ | High |
| vspcampusconnect.vsp.com | United States | | $5000+ | 10,000+ | 1,000+ | High |
| iuware.iu.edu | United States | | $5000+ | 50,000+ | 100+ | Very High |
| sandiegouniontribune.com | United States | | $5000+ | 100+ | | Very High |
| catering.sprouts.com | United States | | $5000+ | 10,000+ | 10,000+ | Medium |
| townofbluffton.sc.gov | United States | | $5000+ | 1,000+ | 100+ | Very High |
| journal-news.com | United States | | $5000+ | 5,000+ | | High |
| beaconbeta.schneidercorp.com | United States | | $1000+ | 250+ | 10,000+ | Medium |
| action.splcenter.org | United States | | $1000+ | 300,000+ | 100+ | Very High |
| sgvtribune.com | United States | | $10000+ | 10,000+ | | High |
| join.me | United States | | $5000+ | 5,000+ | 1+ | High |
| seagate.com | United States | $615m+ | $10000+ | 5,000+ | 10,000+ | Very High |
| irobot.com | United States | $98m+ | $10000+ | 20,000+ | 100+ | High |
| spokesman.com | United States | | $5000+ | 20,000+ | | Very High |
| 3m.com | United States | $3.3m+ | $10000+ | | 10,000+ | Very High |
| jmbullion.com | United States | $165k+ | $5000+ | 2,000+ | 1,000+ | Medium |

https://trends.builtwith.com/websitelist/Microsoft-Azure-CDN/United-States

104. On information and belief, Microsoft Azure CDN utilizes a table to correlate shared repeater servers to origin servers hosting the requested content. For instance, on information and belief, Microsoft Azure CDN includes servers in the edge infrastructure that cache content for multiple Azure CDN customers and which use one or more tables to determine an origin server for a CDN customer.

## How it works




1. A user (Alice) requests a file (also called an asset) by using a URL with a special domain name, such as <*endpoint name*>.azureedge.net. This name can be an endpoint hostname or a custom domain. The DNS routes the request to the best performing POP location, which is usually the POP that is geographically closest to the user.
2. If no edge servers in the POP have the file in their cache, the POP requests the file from the origin server. The origin server can be an Azure Web App, Azure Cloud Service, Azure Storage account, or any publicly accessible web server.
3. The origin server returns the file to an edge server in the POP.
4. An edge server in the POP caches the file and returns the file to the original requestor (Alice). The file remains cached on the edge server in the POP until the time to live (TTL) specified by its HTTP headers expires. If the origin server didn't specify a TTL, the default TTL is seven days.
5. More users can then request the same file by using the same URL that Alice used, and gets directed to the same POP.
6. If the TTL for the file hasn't expired, the POP edge server returns the file directly from the cache. This process results in a faster, more responsive user experience.

https://learn.microsoft.com/en-us/azure/cdn/cdn-overview

# How and why a CDN is used

Typical uses for a CDN include:

- Delivering static resources for client applications, often from a website. These resources can be images, style sheets, documents, files, client-side scripts, HTML pages, HTML fragments, or any other content that the server does not need to modify for each request. The application can create items at runtime and make them available to the CDN (for example, by creating a list of current news headlines), but it does not do so for each request.
- Delivering public static and shared content to devices such as mobile phones and tablet computers. The application itself is a web service that offers an API to clients running on the various devices. The CDN can also deliver static datasets (via the web service) for the clients to use, perhaps to generate the client UI. For example, the CDN could be used to distribute JSON or XML documents.
- Serving entire websites that consist of only public static content to clients, without requiring any dedicated compute resources.
- Streaming video files to the client on demand. Video benefits from the low latency and reliable connectivity available from the globally located datacenters that offer CDN connections. Microsoft Azure Media Services integrates with Azure Content Delivery Network to deliver content directly to the CDN for further distribution. For more information, see Streaming endpoints overview.
- Generally improving the experience for users, especially those located far from the datacenter hosting the application. These users might otherwise suffer higher latency. A large proportion of the total size of the content in a web application is often static, and using the CDN can help to maintain performance and overall user experience while eliminating the requirement to deploy the application to multiple datacenters. For a list of Azure Content Delivery Network node locations, see Azure CDN POP Locations.
- Supporting IoT (Internet of Things) solutions. The huge numbers of devices and appliances involved in an IoT solution could easily overwhelm an application if it had to distribute firmware updates directly to each device.
- Coping with peaks and surges in demand without requiring the application to scale, avoiding the consequent increase in running costs. For example, when an update to an operating system is released for a hardware device such as a specific model of router, or for a consumer device such as a smart TV, there will be a huge peak in demand as it is downloaded by millions of users and devices over a short period.

https://learn.microsoft.com/en-us/azure/architecture/best-practices/cdn

# Edge Nodes - List

Reference

Feedback

Service: CDN

API Version: 2024-02-01

Edgenodes are the global Point of Presence (POP) locations used to deliver CDN content to end users.

HTTP

Copy

Try It

```
GET https://management.azure.com/providers/Microsoft.Cdn/edgenodes?api-version=2024-02-01
```

https://learn.microsoft.com/en-us/rest/api/cdn/edge-nodes/list?view=rest-cdn-2024-02-01&tabs=HTTP

## EdgeNode

Edgenode is a global Point of Presence (POP) location used to deliver CDN content to end users.

Expand table

| Name | Type | Description |
|---|---|---|
| id | string | Resource ID. |
| name | string | Resource name. |
| properties.ipAddressGroups | IpAddressGroup[] | List of ip address groups. |
| systemData | SystemData | Read only system data |
| type | string | Resource type. |

https://learn.microsoft.com/en-us/rest/api/cdn/edge-nodes/list?view=rest-cdn-2024-02-01&tabs=HTTP

## COUNT II: INFRINGEMENT OF THE '466 PATENT

105. Plaintiff hereby incorporates by reference each of the allegations in the foregoing paragraphs as though fully set forth herein, and further alleges as follows.

106. Microsoft infringes at least claim 1 of the '466 Patent through operation of one or more of the Accused Microsoft Functionalities. For instance, Microsoft Azure CDN distributes objects maintained on origin servers via its content delivery network. Microsoft Azure CDN comprises servers including edge servers (e.g., EdgeNodes, Edge POPs) and parent servers (e.g., regional cache POPs) distinct from the origin server.

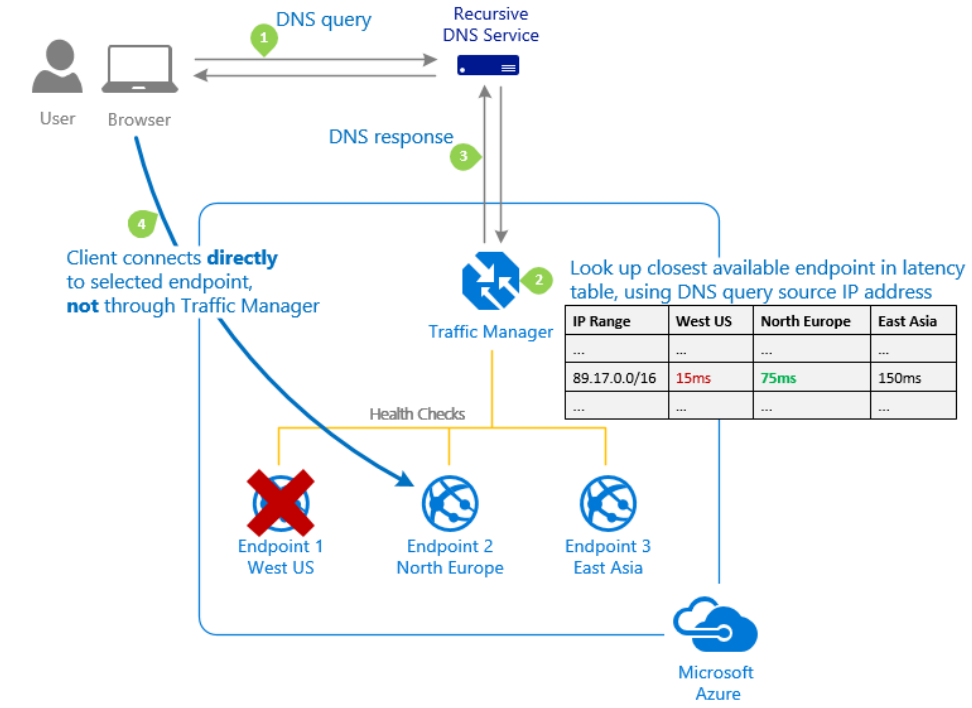
Connectivity within Microsoft's network along with new Regional Caching capabilities enables more consistent, more predictable cache fill performance by providing multi-tier caching along direct, private access to content in Azure from each CDN edge point of presence (POP). Azure CDN from Microsoft entered public preview providing access to 54 global Edge POPs in 33 countries and 16 Regional Cache POPs.


https://azure.microsoft.com/es-es/blog/new-locations-for-azure-cdn-now-available/

107. Microsoft Azure CDN directs client requests to edge nodes selected at least based in part on network traffic conditions.

## Performance traffic-routing method

Deploying endpoints in two or more locations across the globe can improve the responsiveness of your applications. With the 'Performance' traffic-routing method, you can route traffic to the location that is 'closest' to you.




https://learn.microsoft.com/en-us/azure/traffic-manager/traffic-manager-routing-methods

The 'closest' endpoint isn't necessarily closest as measured by geographic distance. Instead, the 'Performance' traffic-routing method determines the closest endpoint by measuring network latency. Traffic Manager maintains an Internet Latency Table to track the round-trip time between IP address ranges and each Azure datacenter.

Traffic Manager looks up the source IP address of the incoming DNS request in the Internet Latency Table. Traffic Manager then chooses an available endpoint in the Azure datacenter that has the lowest latency for that IP address range. Then Traffic Manager returns that endpoint in the DNS response.

As explained in How Traffic Manager Works, Traffic Manager doesn't receive DNS queries directly from clients. Instead, DNS queries come from the recursive DNS service that the clients are configured to use. As such, the IP address used to determine the 'closest' endpoint isn't the client's IP address, but it's the IP address of the recursive DNS service. This IP address is a good proxy for the client.

Traffic Manager regularly updates the Internet Latency Table to account for changes in the global Internet and new Azure regions. However, application performance varies based on real-time variations in load across the Internet. Performance traffic-routing doesn't monitor load on a given service endpoint. If an endpoint becomes unavailable, Traffic Manager won't include it in the DNS query responses.

https://learn.microsoft.com/en-us/azure/traffic-manager/traffic-manager-routing-methods

108. If the first edge server has the requested object, Microsoft serves the requested object to the client from the first edge server. On information and belief, if a first edge server does not have

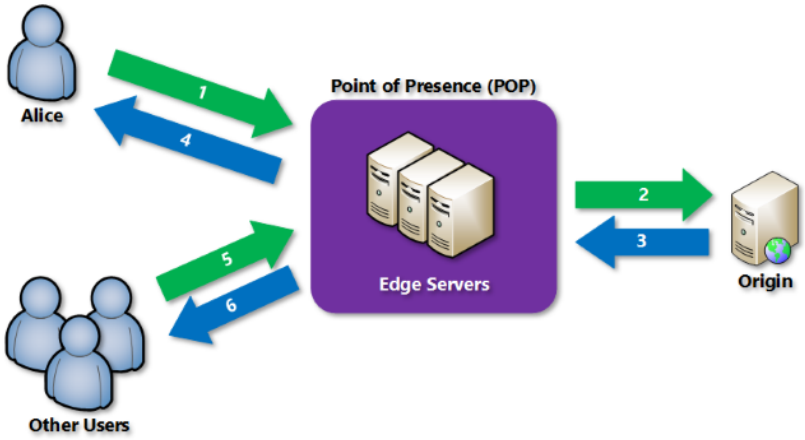
a requested object, the client is redirected to a second server (e.g., a parent POP) in the CDN to handle the request.

| | |
|---|---|
| SentToOriginShield (deprecated) * **See notes on deprecation in the following section.** | If true, it means that request was answered from origin shield cache instead of the edge pop. Origin shield is a parent cache used to improve cache hit ratio. |
| isReceivedFromClient | If true, it means that the request came from the client. If false, the request is a miss in the edge (child POP) and is responded from origin shield (parent POP). |

https://learn.microsoft.com/en-us/azure/cdn/monitoring-and-access-log

109. On further information and belief, if the requested object is not stored on the second server (e.g., regional POP), the client is directed to another server in the CDN (e.g., an origin server) to serve the requested object.

## How it works




1. A user (Alice) requests a file (also called an asset) by using a URL with a special domain name, such as <*endpoint name*>.azureedge.net. This name can be an endpoint hostname or a custom domain. The DNS routes the request to the best performing POP location, which is usually the POP that is geographically closest to the user.
2. If no edge servers in the POP have the file in their cache, the POP requests the file from the origin server. The origin server can be an Azure Web App, Azure Cloud Service, Azure Storage account, or any publicly accessible web server.
3. The origin server returns the file to an edge server in the POP.
4. An edge server in the POP caches the file and returns the file to the original requestor (Alice). The file remains cached on the edge server in the POP until the time to live (TTL) specified by its HTTP headers expires. If the origin server didn't specify a TTL, the default TTL is seven days.
5. More users can then request the same file by using the same URL that Alice used, and gets directed to the same POP.
6. If the TTL for the file hasn't expired, the POP edge server returns the file directly from the cache. This process results in a faster, more responsive user experience.

https://learn.microsoft.com/en-us/azure/cdn/cdn-overview

**COUNT III: INFRINGEMENT OF THE '053 Patent**

110. Plaintiff hereby incorporates by reference each of the allegations in the foregoing paragraphs as though fully set forth herein, and further alleges as follows.

111. Microsoft infringes at least claim 27 of the '053 Patent through operation of one or more of the Accused Microsoft Functionalities. For instance, Azure Front Door provides hardware and software components in a CDN serving resources on behalf of customers of the CDN.

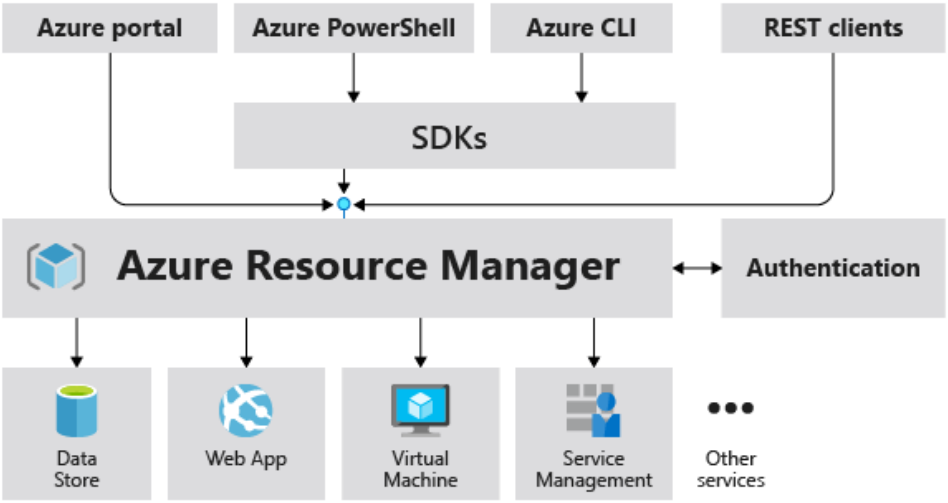
> Azure Front Door is Microsoft's modern cloud Content Delivery Network (CDN) that provides fast, reliable, and secure access between your users and your applications' static and dynamic web content across the globe. Azure Front Door delivers your content using Microsoft's global edge network with hundreds of global and local points of presence (PoPs) distributed around the world close to both your enterprise and consumer end users.
>
> https://learn.microsoft.com/en-us/azure/frontdoor/front-door-overview

112. On information and belief, Azure CDN cache servers obtain global configuration data (e.g., routing policies, caching rules, content updates, and security settings) from a control core such as Azure Resource Manager.

## Consistent management layer

When you send a request through any of the Azure APIs, tools, or SDKs, Resource Manager receives the request. It authenticates and authorizes the request before forwarding it to the appropriate Azure service. Because all requests are handled through the same API, you see consistent results and capabilities in all the different tools.

The following image shows the role Azure Resource Manager plays in handling Azure requests.




All capabilities that are available in the portal are also available through PowerShell, Azure CLI, REST APIs, and client SDKs. Functionality initially released through APIs are represented in the portal within 180 days of initial release.

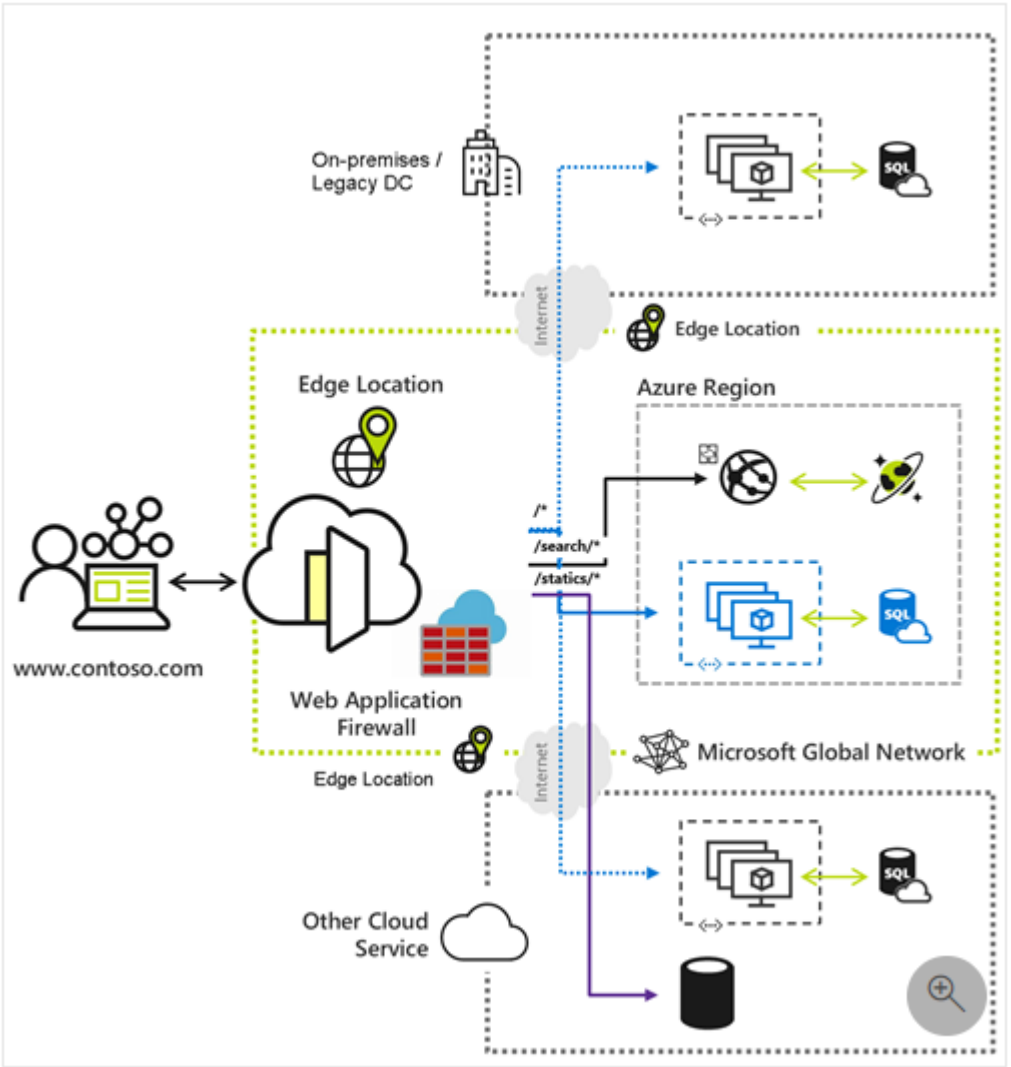
https://learn.microsoft.com/en-us/azure/azure-resource-manager/management/overview

113. As an example, an edge node may obtain updated global routing policies or caching rules as needed.

114. Azure Front Door further provides rule sets configurable specific to CDN customers. On information and belief, Azure CDN serves content according to global configuration (e.g., routing policies, caching rules, content updates, and security settings) and customer configuration (e.g., rule sets) associated with the particular customer.

A rule set is a customized rules engine that groups a combination of rules into a single set. You can associate a rule set with multiple routes. A Rule set allows you to customize how requests get processed and handled at the Azure Front Door edge.

https://learn.microsoft.com/en-us/azure/frontdoor/front-door-rules-engine




https://learn.microsoft.com/en-us/azure/frontdoor/front-door-overview

## Deliver modern apps and architectures

Modernize your internet first applications on Azure with Cloud Native experiences

- Integrate with DevOps friendly command line tools across SDKs of different languages, Bicep, ARM templates, CLI and PowerShell.
- Define your own custom domain with flexible domain validation.
- Load balance and route traffic across origins and use intelligent health probe monitoring across apps or content hosted in Azure or anywhere.
- Integrate with other Azure services such as DNS, Web Apps, Storage and many more for domain and origin management.
- Move your routing business logic to the edge with enhanced rules engine capabilities including regular expressions and server variables.
- Analyze built-in reports with an all-in-one dashboard for both Front Door and security patterns.
- Monitoring your Front Door traffic in real time, and configure alerts that integrate with Azure Monitor.
- Log each Front Door request and failed health probes.

https://learn.microsoft.com/en-us/azure/frontdoor/front-door-overview

**COUNT IV: INFRINGEMENT OF THE '883 PATENT**

115. Plaintiff hereby incorporates by reference each of the allegations in the foregoing paragraphs as though fully set forth herein, and further alleges as follows.

116. Microsoft infringes at least claim 1 of the '883 Patent through operation of one or more of the Accused Microsoft Functionalities. For instance, Azure Service Fabric provides a network comprising multiple service endpoints (e.g., clusters) running on a plurality of devices (e.g., virtual machine servers) comprising hardware including memory and at least one processor.

Azure Service Fabric is a distributed systems platform that makes it easy to package, deploy, and manage scalable and reliable microservices. Service Fabric is a container and process orchestrator that allows you to host your clusters anywhere: on Azure, in an on-premises datacenter, or on any cloud provider. You can use any framework to write your services and choose where to run the application from multiple environment choices. This article details the terminology used by Service Fabric to understand the terms used in the documentation.

https://learn.microsoft.com/en-us/azure/service-fabric/service-fabric-technical-overview

117. Azure Service Fabric clusters assume one or more discrete responsibilities (e.g., microservices) in order to collaboratively process requests across the group of service endpoints.

# Infrastructure concepts

**Cluster**: A network-connected set of virtual or physical machines into which your microservices are deployed and managed. Clusters can scale to thousands of machines.

**Node**: A machine or VM that's part of a cluster is called a *node*. Each node is assigned a node name (string). Nodes have characteristics, such as placement properties. Each machine or VM has an auto-start Windows service, `FabricHost.exe`, that starts running upon boot and then starts two executables: `Fabric.exe` and `FabricGateway.exe`. These two executables make up the node. For testing scenarios, you can host multiple nodes on a single machine or VM by running multiple instances of `Fabric.exe` and `FabricGateway.exe`.

https://learn.microsoft.com/en-us/azure/service-fabric/service-fabric-technical-overview

# Application and service concepts

**Service Fabric Native Application**: Service Fabric Native Applications are described by the Native Application Model (XML-based application and service manifests).

## Service Fabric Native Application concepts

**Application**: An application is a collection of constituent services that perform a certain function or functions. The lifecycle of each application instance can be managed independently.

**Service**: A service performs a complete and standalone function and can start and run independently of other services. A service is composed of code, configuration, and data. For each service, code consists of the executable binaries, configuration consists of service settings that can be loaded at run time, and data consists of arbitrary static data to be consumed by the service.

**Application type**: The name/version assigned to a collection of service types. It is defined in an `ApplicationManifest.xml` file and embedded in an application package directory. The directory is then copied to the Service Fabric cluster's image store. You can then create a named application from this application type within the cluster.

https://learn.microsoft.com/en-us/azure/service-fabric/service-fabric-technical-overview

118. Azure Policy features configurable definitions for Azure Service Fabric such as cluster redundancy and regional deployment including across different cloud providers using Azure Arc.

> Azure Policy helps to enforce organizational standards and to assess compliance at-scale. Through its compliance dashboard, it provides an aggregated view to evaluate the overall state of the environment, with the ability to drill down to the per-resource, per-policy granularity. It also helps to bring your resources to compliance through bulk remediation for existing resources and automatic remediation for new resources.

https://learn.microsoft.com/en-us/azure/governance/policy/overview

119. Azure Service Fabric cluster nodes receive requests for resources (e.g., services provided by an application). For instance, Azure Service Fabric nodes may be responsible for system services (e.g., name resolution, service discovery), application hosting (e.g., handling client requests), primary and non-primary workloads, resource management (e.g., scaling, load balancing), or health monitoring.

# Application and service concepts

**Service Fabric Native Application**: Service Fabric Native Applications are described by the Native Application Model (XML-based application and service manifests).

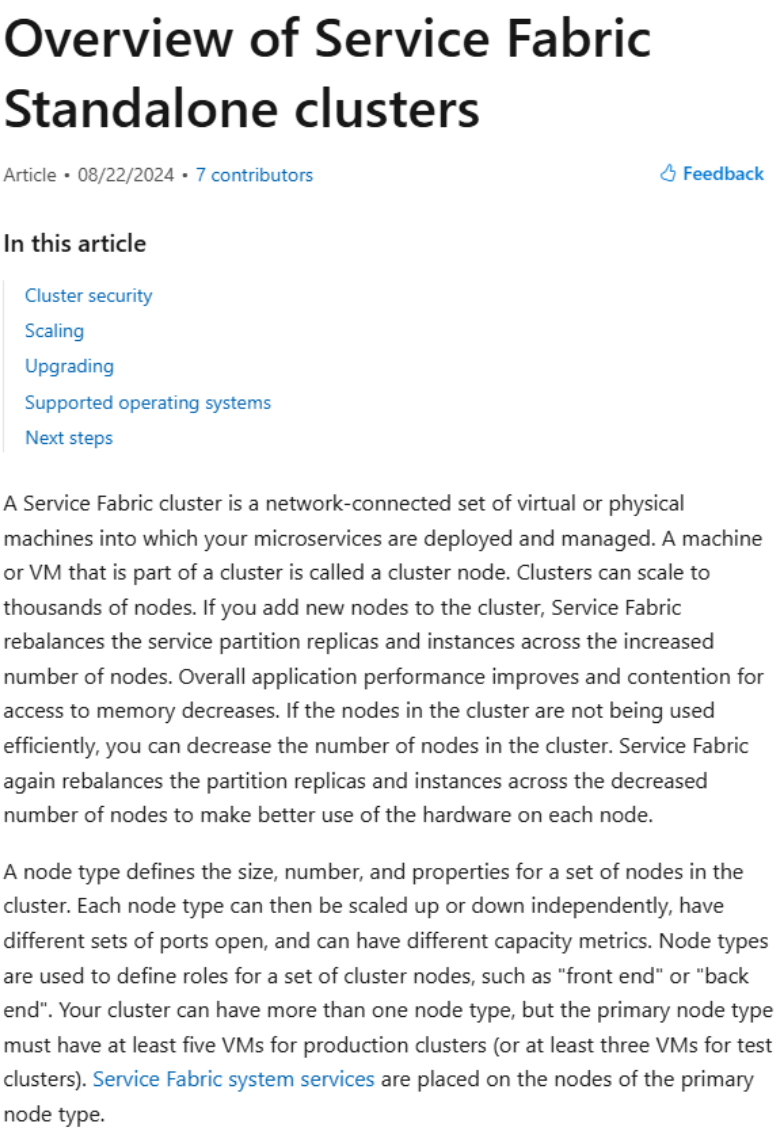
## Service Fabric Native Application concepts

**Application**: An application is a collection of constituent services that perform a certain function or functions. The lifecycle of each application instance can be managed independently.

**Service**: A service performs a complete and standalone function and can start and run independently of other services. A service is composed of code, configuration, and data. For each service, code consists of the executable binaries, configuration consists of service settings that can be loaded at run time, and data consists of arbitrary static data to be consumed by the service.

**Application type**: The name/version assigned to a collection of service types. It is defined in an `ApplicationManifest.xml` file and embedded in an application package directory. The directory is then copied to the Service Fabric cluster's image store. You can then create a named application from this application type within the cluster.

https://learn.microsoft.com/en-us/azure/service-fabric/service-fabric-technical-overview



# Overview of Service Fabric Standalone clusters

Article • 08/22/2024 • 7 contributors

Feedback

In this article

- Cluster security
- Scaling
- Upgrading
- Supported operating systems
- Next steps

A Service Fabric cluster is a network-connected set of virtual or physical machines into which your microservices are deployed and managed. A machine or VM that is part of a cluster is called a cluster node. Clusters can scale to thousands of nodes. If you add new nodes to the cluster, Service Fabric rebalances the service partition replicas and instances across the increased number of nodes. Overall application performance improves and contention for access to memory decreases. If the nodes in the cluster are not being used efficiently, you can decrease the number of nodes in the cluster. Service Fabric again rebalances the partition replicas and instances across the decreased number of nodes to make better use of the hardware on each node.

A node type defines the size, number, and properties for a set of nodes in the cluster. Each node type can then be scaled up or down independently, have different sets of ports open, and can have different capacity metrics. Node types are used to define roles for a set of cluster nodes, such as "front end" or "back end". Your cluster can have more than one node type, but the primary node type must have at least five VMs for production clusters (or at least three VMs for test clusters). Service Fabric system services are placed on the nodes of the primary node type.

https://learn.microsoft.com/en-us/azure/service-fabric/service-fabric-technical-overview

120. Requests for resources are processed in accordance with the type of resource and the kind of responsibility a node has for that particular resource.

## Key features and benefits

Some of the key scenarios that Azure Arc supports are:

- Implement consistent inventory, management, governance, and security for servers across your environment.
- Configure Azure VM extensions to use Azure management services to monitor, secure, and update your servers.
- Manage and govern Kubernetes clusters at scale.
- Use GitOps to deploy configurations across one or more clusters from Git repositories.
- Zero-touch compliance and configuration for Kubernetes clusters using Azure Policy.
- Run Azure data services on any Kubernetes environment as if it runs in Azure (specifically Azure SQL Managed Instance and Azure Database for PostgreSQL server, with benefits such as upgrades, updates, security, and monitoring). Use elastic scale and apply updates without any application downtime, even without continuous connection to Azure.
- Create custom locations on top of your Azure Arc-enabled Kubernetes clusters, using them as target locations for deploying Azure services instances. Deploy your Azure service cluster extensions for Azure Arc-enabled data services, App services on Azure Arc (including web, function, and logic apps) and Event Grid on Kubernetes.
- Perform virtual machine lifecycle and management operations for VMware vSphere and Azure Stack HCI environments.
- A unified experience viewing your Azure Arc-enabled resources, whether you are using the Azure portal, the Azure CLI, Azure PowerShell, or Azure REST API.

https://learn.microsoft.com/en-us/azure/azure-arc/overview

**COUNT V: INFRINGEMENT OF THE '692 PATENT**

121. Plaintiff hereby incorporates by reference each of the allegations in the foregoing paragraphs as though fully set forth herein, and further alleges as follows.

122. Microsoft infringes at least claim 1 of the '692 Patent through operation of one or more Accused Microsoft Functionalities. For instance, Azure CDN servers obtain requests for resources available as part of a content provider's library.

# Introduction to caching

Caching is the process of storing data locally so that future requests for that data can be accessed more quickly. In the most common type of caching, web browser caching, a web browser stores copies of static data locally on a local hard drive. By using caching, the web browser can avoid making multiple round-trips to the server and instead access the same data locally, thus saving time and resources. Caching is well-suited for locally managing small, static data such as static images, CSS files, and JavaScript files.

Similarly, caching is used by a content delivery network on edge servers close to the user to avoid requests traveling back to the origin and reducing end-user latency. Unlike a web browser cache, which is used only for a single user, the content delivery network has a shared cache. In a content delivery network shared cache, a file request by a user can be used by another user, which greatly decreases the number of requests to the origin server.

Dynamic resources that change frequently or are unique to an individual user can't be cached. Those types of resources, however, can take advantage of dynamic site acceleration (DSA) optimization on the Azure content delivery network for performance improvements.

https://learn.microsoft.com/en-us/azure/cdn/cdn-how-caching-works

123. Azure CDN servers may evict content stored in a network cache if the content is less frequently requested (e.g., less popular). If the resource is not available at the first server or at a peer of the first server, on information and belief, Microsoft determines if the resource is popular—for example, based on cache eviction policies. If the resource is determined to be popular, then Microsoft obtains the resource and serves the resource to the client via the first server.

> ⓘ Note
>
> Azure Content Delivery Network makes no guarantees about minimum amount of time that the object will be stored in the cache. Cached contents might be evicted from the content delivery network cache before they are expired if the contents are not requested as frequently to make room for more frequently requested contents.

https://learn.microsoft.com/en-us/azure/cdn/cdn-how-caching-works

124. On information and belief, if the resource is determined to not be popular, Microsoft directs the client to a second server in a second tier of servers distinct from the first tier of servers. On information and belief, Microsoft's second server comprises a first portion of the content provider's library, the first portion comprising at least the resource, wherein at least one other server in the second tier comprises a second portion of the content provider's library. On information and belief, the first portion of the content provider's library is distinct from the second portion. On information and belief, the second tier is any intermediate tier between the first tier and an origin server that stores resources associated with the content provider's library. On information and belief, Microsoft's second server serves the resource to the client.

> Caching can occur at multiple levels between the origin server and the end user:
>
> - Web server: Uses a shared cache (for multiple users).
> - Content delivery network: Uses a shared cache (for multiple users).
> - Internet service provider (ISP): Uses a shared cache (for multiple users).
> - Web browser: Uses a private cache (for one user).
>
> Each cache typically manages its own resource freshness and performs validation when a file is stale. This behavior is defined in the HTTP caching specification, RFC 7234.

https://learn.microsoft.com/en-us/azure/cdn/cdn-how-caching-works

## Resource freshness

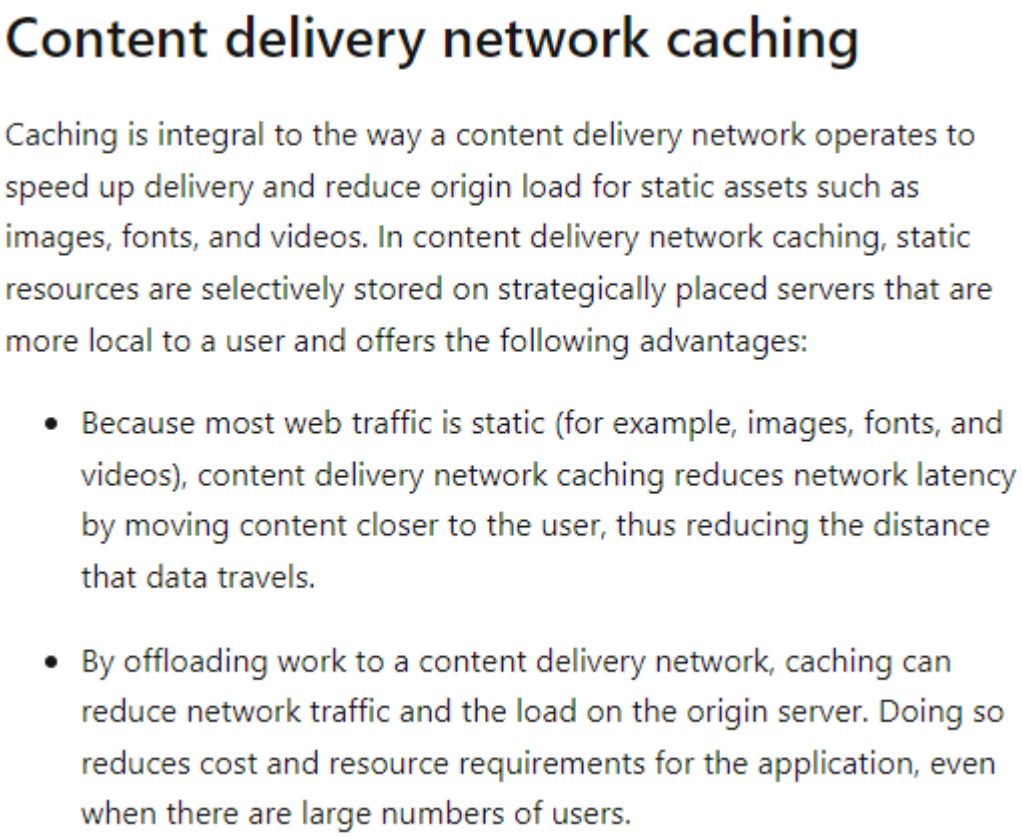
Since a cached resource can potentially be out-of-date, or stale (as compared to the corresponding resource on the origin server), it's important for any caching mechanism to control when content gets a refresh. To save time and bandwidth consumption, a cached resource isn't compared to the version on the origin server every time it's accessed. Instead, as long as a cached resource is considered to be fresh, it's assumed to be the most current version and is sent directly to the client. A cached resource is considered to be fresh when its age is less than the age or period defined by a cache setting. For example, when a browser reloads a web page, it verifies that each cached resource on your hard drive is fresh and loads it. If the resource isn't fresh (stale), an up-to-date copy is loaded from the server.

https://learn.microsoft.com/en-us/azure/cdn/cdn-how-caching-works

## Validation

If a resource is considered stale, the origin server gets asked to validate it to determine whether the data in the cache still matches what's on the origin server. If the file has been modified on the origin server, the cache updates its version of the resource. Otherwise, if the resource is fresh, the data is delivered directly from the cache without validating it first.

https://learn.microsoft.com/en-us/azure/cdn/cdn-how-caching-works




https://learn.microsoft.com/en-us/azure/cdn/cdn-how-caching-works

**COUNT VI: INFRINGEMENT OF THE '173 PATENT**

125. Plaintiff hereby incorporates by reference each of the allegations in the foregoing paragraphs as though fully set forth herein, and further alleges as follows.

126. Microsoft infringes at least claim 1 of the '173 Patent through operation of one or more of the Accused Microsoft Functionalities. For instance, Microsoft Azure CDN comprises nodes (e.g., EdgeNodes or Endpoints) with locally cached content as well as policy-based caching and content versioning. For instance, Microsoft Azure CDN offers content caching rules which determine time periods for expiration of content in the cache.

Azure Content Delivery Network offers two ways to control how your files get cached:

**Caching rules:** Azure Content Delivery Network provides two types of caching rules: global and custom.

- Global caching rules - You can set one global caching rule for each endpoint in your profile, which affects all requests to the endpoint. The global caching rule overrides any HTTP cache-directive headers, if set.
- Custom caching rules - You can set one or more custom caching rules for each endpoint in your profile. Custom caching rules match specific paths and file extensions, get processed in order, and override the global caching rule, if set.

**Query string caching:** You can adjust how the Azure content delivery network treats caching for requests with query strings. For information, see Control Azure Content Delivery Network caching behavior with query strings. If the file isn't cacheable, the query string caching setting has no effect, based on caching rules and content delivery network default behaviors.

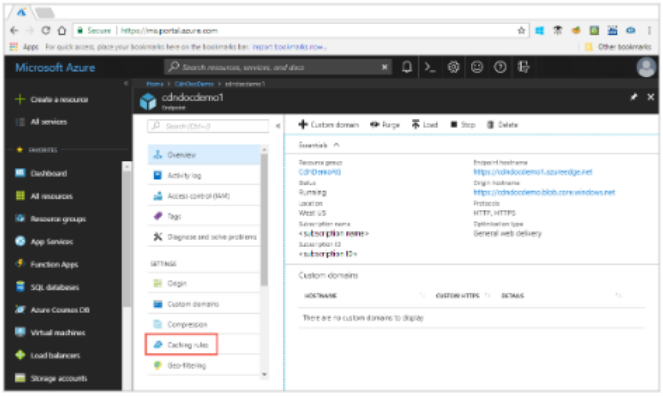
For information about default caching behavior and caching directive headers, see How caching works.

https://learn.microsoft.com/en-us/azure/cdn/cdn-caching-rules

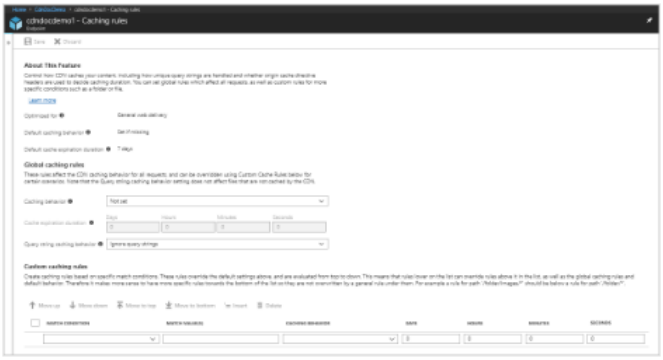
# Accessing Azure Content Delivery Network caching rules

1. Open the Azure portal, select a content delivery network profile, then select an endpoint.

2. In the left pane under Settings, select **Caching rules**.




The **Caching rules** page appears.




https://learn.microsoft.com/en-us/azure/cdn/cdn-caching-rules

# Caching behavior settings

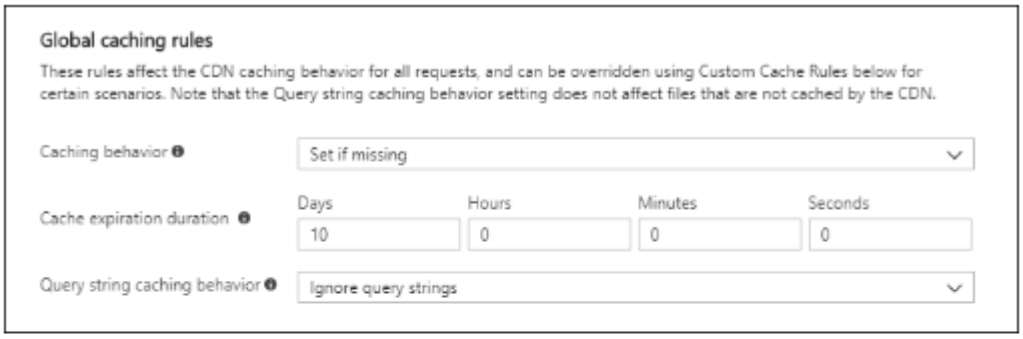
For global and custom caching rules, you can specify the following **Caching behavior** settings:

- **Bypass cache:** Don't cache and ignore origin-provided cache-directive headers.
- **Override:** Ignore origin-provided cache duration; use the provided cache duration instead. This setting doesn't override cache-control: no-cache.

> ⓘ Note
>
> For **Azure CDN from Microsoft** profiles, cache expiration override is only applicable to status codes 200 and 206.

- **Set if missing:** Honor origin-provided cache-directive headers, if they exist; otherwise, use the provided cache duration.

https://learn.microsoft.com/en-us/azure/cdn/cdn-caching-rules





https://learn.microsoft.com/en-us/azure/cdn/cdn-caching-rules

# Cache expiration duration

For global and custom caching rules, you can specify the cache expiration duration in days, hours, minutes, and seconds:

- For the **Override** and **Set if missing Caching behavior** settings, valid cache durations range between 0 seconds and 366 days. For a value of 0 seconds, the content delivery network caches the content, but must revalidate each request with the origin server.
- For the **Bypass cache** setting, the cache duration gets automatically set to 0 seconds, which isn't a modifiable value.

https://learn.microsoft.com/en-us/azure/cdn/cdn-caching-rules

# Custom caching rules match conditions

For custom cache rules, two match conditions are available:

- **Path:** This condition matches the path of the URL, excluding the domain name, and supports the wildcard symbol (*). For example, */myfile.html*, */my/folder/***, *and /my/images/.jpg*. The maximum length is 260 characters.

- **Extension:** This condition matches the file extension of the requested file. You can provide a list of comma-separated file extensions to match. For example, *.jpg*, *.mp3*, or *.png*. The maximum number of extensions is 50 and the maximum number of characters per extension is 16.

https://learn.microsoft.com/en-us/azure/cdn/cdn-caching-rules

# Global and custom rule processing order

Global and custom caching rules get processed in the following order:

- Global caching rules take precedence over the default content delivery network caching behavior (HTTP cache-directive header settings).
- Custom caching rules take precedence over global caching rules, where they apply. Custom caching rules get processed in order from top to bottom. That is, if a request matches both conditions, rules at the bottom of the list take precedence over rules at the top of the list. Therefore, you should place more specific rules lower in the list.

**Example**:

- Global caching rule:
  - Caching behavior: **Override**
  - Cache expiration duration: One day
- Custom caching rule #1:
  - Match condition: **Path**
  - Match value: */home/**
  - Caching behavior: **Override**
  - Cache expiration duration: Two days
- Custom caching rule #2:
  - Match condition: **Extension**
  - Match value: *.html*
  - Caching behavior: **Set if missing**
  - Cache expiration duration: Three days

https://learn.microsoft.com/en-us/azure/cdn/cdn-caching-rules

When you set these rules, a request for <*endpoint hostname*>.azureedge.net/home/index.html triggers custom caching rule #2, which get set to: **Set if missing** and 3 days. Therefore, if the *index.html* file has `Cache-Control` or `Expires` HTTP headers, they get honored; otherwise, if you don't set these headers, the file gets cached for three days.

> ⓘ **Note**
>
> Files that are cached before a rule change maintain their origin cache duration setting. To reset their cache durations, you must purge the file.
>
> Azure Content Delivery Network configuration changes can take some time to propagate through the network:
>
> - For **Azure CDN Standard from Edgio** profiles, propagation usually completes in 10 minutes.

https://learn.microsoft.com/en-us/azure/cdn/cdn-caching-rules

127. Based on the associated caching rule, Microsoft Azure CDN nodes determine whether it is acceptable to serve the requested content. For instance, based on the cache policy and file validation, Microsoft Azure CDN nodes may obtain a new version of content to be served.

# Introduction to caching

Caching is the process of storing data locally so that future requests for that data can be accessed more quickly. In the most common type of caching, web browser caching, a web browser stores copies of static data locally on a local hard drive. By using caching, the web browser can avoid making multiple round-trips to the server and instead access the same data locally, thus saving time and resources. Caching is well-suited for locally managing small, static data such as static images, CSS files, and JavaScript files.

Similarly, caching is used by a content delivery network on edge servers close to the user to avoid requests traveling back to the origin and reducing end-user latency. Unlike a web browser cache, which is used only for a single user, the content delivery network has a shared cache. In a content delivery network shared cache, a file request by a user can be used by another user, which greatly decreases the number of requests to the origin server.

Dynamic resources that change frequently or are unique to an individual user can't be cached. Those types of resources, however, can take advantage of dynamic site acceleration (DSA) optimization on the Azure content delivery network for performance improvements.

https://learn.microsoft.com/en-us/azure/cdn/cdn-how-caching-works

Caching can occur at multiple levels between the origin server and the end user:

- Web server: Uses a shared cache (for multiple users).
- Content delivery network: Uses a shared cache (for multiple users).
- Internet service provider (ISP): Uses a shared cache (for multiple users).
- Web browser: Uses a private cache (for one user).

Each cache typically manages its own resource freshness and performs validation when a file is stale. This behavior is defined in the HTTP caching specification, RFC 7234 .

https://learn.microsoft.com/en-us/azure/cdn/cdn-how-caching-works

## Resource freshness

Since a cached resource can potentially be out-of-date, or stale (as compared to the corresponding resource on the origin server), it's important for any caching mechanism to control when content gets a refresh. To save time and bandwidth consumption, a cached resource isn't compared to the version on the origin server every time it's accessed. Instead, as long as a cached resource is considered to be fresh, it's assumed to be the most current version and is sent directly to the client. A cached resource is considered to be fresh when its age is less than the age or period defined by a cache setting. For example, when a browser reloads a web page, it verifies that each cached resource on your hard drive is fresh and loads it. If the resource isn't fresh (stale), an up-to-date copy is loaded from the server.

https://learn.microsoft.com/en-us/azure/cdn/cdn-how-caching-works

**PRAYER FOR RELIEF**

Wherefore, Plaintiff requests entry of judgment in its favor and against Microsoft as follows:

A. Judgment that Microsoft has directly infringed one or more claims of each of the Asserted Patents pursuant to 35 U.S.C. § 271;

B. An award of lost profits and/or reasonable royalty damages to compensate Plaintiff for Microsoft's infringement, including damages pursuant to 35 U.S.C. § 284, as well as prejudgment and post-judgment interest;

C. An award of costs and expenses in this action, including an award of Plaintiff's reasonable attorneys' fees pursuant to 35 U.S.C. § 285;

D. A permanent injunction restraining and enjoining Microsoft, and its respective officers, agents, servants, employees, attorneys, and those persons in active concert or participation with Microsoft who receive actual

notice of the order by personal service or otherwise, from any further sales or use of their infringing products and/or services and any other infringement of the Asserted Patents;

E. A finding that Microsoft has willfully infringed one or more claims of the Asserted Patents;

F. A finding that this is an exceptional case, and awarding treble damages due to Microsoft's deliberate and willful conduct, and ordering Microsoft to pay Plaintiff's costs of suit and attorneys' fees; and

G. Any such other and further relief as the Court may deem just, proper, and equitable under the circumstances.

**JURY DEMAND**

Plaintiff respectfully demands a trial by jury on all claims and issues so triable.

Dated: June 26, 2025

Respectfully submitted,

*/s/ Ryan D. Dykal*
Ryan D. Dykal
Missouri State Bar No. 60905
Mark D. Schafer (*pro hac vice* forthcoming)
Phillip Eckert (*pro hac vice* forthcoming)
**BOIES SCHILLER FLEXNER LLP**
1401 New York Ave, NW
Washington, DC 20005
(t): 202-274-1109
rdykal@bsfllp.com
mschafer@bsfllp.com
peckert@bsfllp.com

***Counsel for Plaintiff***
***Sandpiper CDN, LLC***