

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

Sandpiper CDN, LLC,

Plaintiff,

v.

Microsoft Corporation,

Defendant.

Civil Case No.

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:

INTRODUCTION

1. This case involves Microsoft’s willful infringement of Sandpiper CDN’s patents. Sandpiper’s patents, which cover foundational aspects of content delivery network (“CDN”) and other networking technologies, enable Microsoft’s \$100 billion-per-year Azure business¹ and other wildly successful Microsoft business units.

2. In June 2025, Sandpiper CDN sued Microsoft in the Eastern District of Texas alleging that Microsoft willfully infringes six patents core to CDN and microservices technologies through operation of Microsoft’s own CDNs for itself and through the paid Azure CDN service it

¹ See “Press Release & Webcast; Earnings Release FY26 Q4” July 29, 2026 <https://www.microsoft.com/en-us/investor/earnings/fy-2026-q4/press-release-webcast> (“This year, Azure revenue surpassed \$100 billion for the first time. . .”).

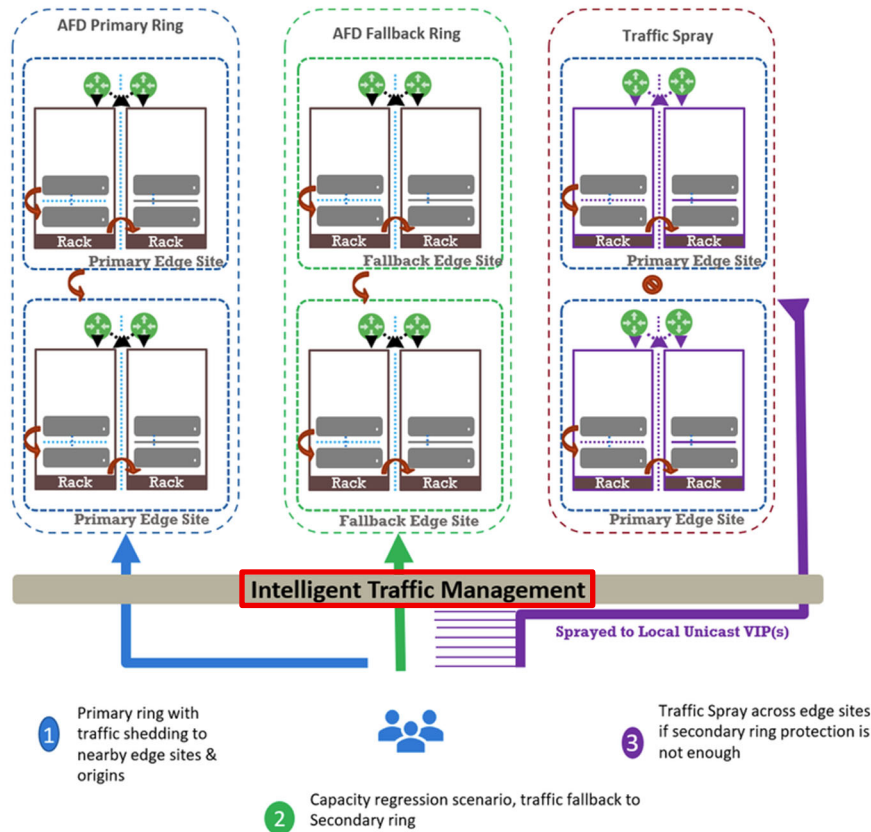
sells to others. *Sandpiper CDN, LLC v. Microsoft*, No. 25-664-JRG-RSP, Dkt. 1 (E.D. Tex. June 24, 2025) (hereinafter, “*Microsoft I*”). In *Microsoft I*, Sandpiper CDN explained how, in 2006, Microsoft offered tens of millions of dollars for a deal in which Microsoft would have acquired all of the patents now owned by Sandpiper CDN, but when the bidding reached \$135 million, Microsoft decided to pay nothing and use the patented technology anyway. *See Microsoft I*, Dkt. 41, ¶¶ 47-53.

3. Microsoft’s willful infringement extends to the patents in this case by other, distinct Azure technologies. For example, while *Microsoft I* did not include patents from the Intelligent Traffic Manager (“ITM”) family of patents, this case does, and these ITM patents were of particular interest to Microsoft even after Microsoft’s failed bid in 2006 to acquire the entire patent portfolio now held by Sandpiper CDN. Indeed, Level 3, after paying \$135 million for these patents, operated its own ITM service, explaining that “Intelligent Traffic Manager (ITM) is a DNS-based, web administered, global load balancing service,” and that Level 3 practices ITM patents in providing its services.² A decade later, Microsoft decided to copy Sandpiper’s ITM product and, by at least 2017, released its own “Intelligent Traffic Manager,” explaining “Azure Traffic Manager is a global load balancer” that “uses the Domain Name System (DNS) protocol.”³ And to this day, Microsoft

² See Intelligent Traffic Manager (ITM) Global Load Balancing Service <https://assets.lumen.com/is/content/Lumen/Service-Guide-ITM-Services-20140423pdf?Creativeid=f6f344ff-d8df-4717-8323-8a2039ac51fa>; *see also* Level 3, “Legal,” <https://web.archive.org/web/20141012113846/http://www.level3.com/en/legal/> (“Certain Level 3 content delivery services are provisioned using systems, methods, and/or functionality covered by one or more of the following U.S. Patents: ... 7,822,871”).

³ See “Microsoft CEO Mentions Containers and Serverless Computing in Keynote,” Sept. 25, 2017, <https://www.sdxcentral.com/news/microsoft-ceo-mentions-containers-and-serverless-computing-in-keynote/> (“As for serverless computing, Microsoft last month added an intelligent traffic manager to its plans with the unveiling of its Azure Event Grid routing service.”); “Architecture best practices for Azure Traffic Manager,” <https://learn.microsoft.com/en-us/azure/well-architected/service-guides/azure-traffic-manager>; “Azure Front Door: Implementing lessons learned following October outages,” <https://techcommunity.microsoft.com/blog/azurenetworkingblog/azure-front-door-implementing-lessons-learned-following-october-outages/4479416> (“We have a highly distributed resilient architecture, which protects against failures at the server, rack, site and even at the regional level. This resiliency is achieved by the use of our intelligent traffic management layer ...”).

refers to its Azure Traffic Manager interchangeably as “Intelligent Traffic Management” and depicts it architecturally in public-facing documents as its own architecture:⁴



4. As explained below, Microsoft also learned of the ITM patent family and its application to CDN services provided by Microsoft at least through public filings in *Microsoft I*.

5. In addition, there are multiple other infringing Microsoft technologies distinct from those accused of infringement in *Microsoft I*, as described below. Despite all of the above, over a year after the filing of *Microsoft I*, Microsoft still has not ceased infringing or paid for a license to any of the patents Sandpiper CDN owns and that Microsoft tried, and failed, to acquire rights to before Microsoft began its infringement. Sandpiper CDN thus files this second lawsuit to hold

⁴ See “Azure Front Door: Implementing lessons learned following October outages” Dec. 18, 2025 <https://techcommunity.microsoft.com/blog/azurenetworkingblog/azure-front-door-implementing-lessons-learned-following-october-outages/4479416>.

Microsoft accountable for its continuing and willful infringement of different Sandpiper CDN patents covering distinct but vitally important Microsoft technologies.

NATURE OF THE ACTION

6. 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. 9,021,112, 7,822,871, 7,860,964, 10,116,738, 8,645,517, 9,660,876, 9,647,899, and 8,156,066 (collectively, “the Asserted Patents”). A true and correct copy of each Asserted Patent is attached to this Complaint as **Exhibits A-H**. 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

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

8. 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

9. 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).

10. 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.

11. 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.

12. 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.

13. 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.⁵

14. 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.”⁶ These locations are operated by Microsoft within Best Buy stores and are regular and established places of business for

⁵ Microsoft, *Microsoft U.S. Office Locations* (available at <https://www.microsoft.com/en-us/about/officelocator/all-offices?msockid=325270f10c6b61543e3b64490de760b4>).

⁶ 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/>).

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.”⁷ 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.⁸

15. Further, Microsoft owns and maintains millions of dollars of real property in Collin County, located within this District.⁹

16. 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.¹⁰

17. 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.

18. Microsoft’s website confirms that its “Azure Content Delivery Network” has a “point of presence (POP) location[]” in “Plano, TX, USA.”¹¹

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

20. Compounding its significant connections to this District, in 2023 Microsoft

⁷ *Id.*

⁸ 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/>).

⁹ See Collin Central Appraisal District, *Property Search* (available at <https://esearch.collincad.org>) (search results for “Microsoft”).

¹⁰ See Denton Central Appraisal District, *Property Search* (available at <https://www.dentoncad.com/property-search>) (search results for “Microsoft”).

¹¹ Microsoft, “Azure Content Delivery Network Coverage by Metro,” Sept. 27, 2024 (available at <https://learn.microsoft.com/en-us/azure/cdn/cdn-pop-locations>).

announced a multi-billion-dollar deal with specialist cloud computing provider CoreWeave to use its datacenters for some of Microsoft's Azure AI workloads.¹² This deal will include Microsoft's use of CoreWeave's \$1.6 billion datacenter in Plano, Texas, located in this District.¹³ Microsoft is currently CoreWeave's biggest customer.¹⁴

21. Courts have repeatedly found jurisdiction over Microsoft to be proper in the Eastern District of Texas, including in numerous patent disputes where Microsoft consented to jurisdiction for or litigated in this District, including in *Microsoft I*.¹⁵

22. 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.

23. 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

¹² 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/>).

¹³ 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/>).

¹⁴ 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/>).

¹⁵ 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).

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.

24. Venue is also proper and convenient in this District because Sandpiper's patent portfolio has been litigated in this District, including specific patents asserted in this case. 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 '876 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) (hereinafter, "*Comcast*"). Comcast filed a motion to transfer, which was denied, and then filed for *mandamus* with the Federal Circuit seeking to set aside denial of transfer, which was also denied.¹⁶ The case proceeded through claim construction, fact discovery, and expert reports, settling during expert discovery.¹⁷

SANDPIPER NETWORKS REVOLUTIONIZES NETWORKING TECHNOLOGIES
VITAL FOR CONTENT DELIVERY

25. 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.

26. 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 server in the 1990s was only capable of handling a limited number of simultaneous connections before

¹⁶ *Id.*, Dkt. 67, 82; see also *in re Comcast Cable Commc'ns, LLC, et al.*, 26-104 (Fed. Cir. Dec. 9, 2025) available at https://fedcircuitblog.com/wp-content/uploads/2025/11/26-104_Comcast_Order.pdf

¹⁷ *Comcast*, Dkt. 113, 134 (May 4, 2026).

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.

27. 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.

28. 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.

29. 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.

30. 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.

31. 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.

32. Sandpiper Networks continued gaining widespread notoriety and won numerous awards for its CDN products and services during this time.¹⁸

33. 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.¹⁹ Digital Island then filed additional patent applications directed to and complementary to CDN technology.

34. Following a series of acquisitions, on or about 2004, the assets of Digital Island and the Sandpiper Networks' patents were purchased by the data center company Savvis Inc. ("Savvis").

¹⁸ 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>.

¹⁹ 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>.

35. 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").

36. 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.

37. Level 3 was acquired by CenturyLink on or about November 2017, and the combined company rebranded as Lumen Technologies, or simply "Lumen," in September 2020.²⁰ Lumen is headquartered in Monroe, Louisiana, which is approximately a two-hour drive from Marshall, Texas.

38. Sandpiper CDN, a successor to Sandpiper Networks, now 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. Like Sandpiper Networks, Sandpiper CDN was co-founded and is co-owned by Mr. Swart who, in addition to licensing its foundational patent portfolio, continues to seek commercialization of the Footprint CDN software as well.²¹

MICROSOFT'S INFRINGEMENT OF THE ASSERTED PATENTS

39. 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.

40. The misappropriation of Sandpiper Networks' patents capitalized on, and

²⁰ 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>.

²¹ See generally, <https://sandpiper-cdn.com>.

undermined, the significant financial investment and years of research into CDN development and implementation undertaken by Sandpiper Networks and its successors.

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

42. 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, including ITM services, using the technology underlying its patents, including one or more of the Asserted Patents at this time.

43. 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.²² 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.

44. 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.

45. 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

²² Jeff Cohen Keynote Address Slide Deck, Content Delivery Summit (May 11, 2009), <https://conferences.infotoday.com/documents/83/CDNSummit09-Keynote-Microsoft.pdf>.

to choose third parties to provide content delivery through the Azure CDN network.²³

46. 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, Microsoft still required third-party CDNs for approximately 75% of its content delivery, including increased reliance on Level 3 and the Asserted Patents.²⁴

47. 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.²⁵

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

49. 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, as well as other Azure products and 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 WILLFULLY INFRINGES THEM

50. 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.

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

²³ 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>.

²⁴ 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>.

²⁵ "Announcing Microsoft's own Content Delivery Network," *Microsoft Azure Blog* (May 7, 2018), [Announcing Microsoft's own Content Delivery Network | Microsoft Azure Blog](https://azure.microsoft.com/en-us/blog/announcing-microsofts-own-content-delivery-network/).

52. 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 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.²⁶

53. In December 2006, Level 3 announced its acquisition of the Savvis CDN business, including the patents stemming from the inventions of Sandpiper Networks, for \$135 million.²⁷ Reports at the time said that there were multiple bidders for these assets.²⁸ Indeed, publicly filed documents describe Level 3 as "the winning bidder" for these patents.²⁹ The unredacted portions of a redacted publicly filed document explains that Level 3 and Limelight both bid on these patents, along with "various third parties," noting "the substantial interest in the CDN patents shown by a number of companies" and the existence of "third-party bids."³⁰ The unredacted portions of another such document notes the "competition [Level 3] faced in making the purchase [of the patents]," the "bids and expressions of interest Limelight and other parties made on the entire Savvis CDN

²⁶ Tech Monitor, "Digital Island Merges with Sandpiper Networks" (Oct. 15, 1999), https://www.techmonitor.ai/hardware/digital_island_merges_with_sandpiper_networks.

²⁷ 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>.

²⁸ Gigaom, "Is Savvis CDN Business For Sale?," aom/is-savvis-cdn-business-for-sale/ (noting "[t]here have been a few bids from interested parties," that the "bids ... are said to be over \$100 million" and that "a majority of its CDN business comes from Microsoft"); *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>.

²⁹ *Level 3 Communications, LLC v. Limelight Networks, Inc.*, 09-589, (hereinafter, "Limelight Case"), Dkt. 1 (D. Del. Dec. 17, 2007).

³⁰ Limelight Case, Dkt. 269, pp. 6-10.

business,” and that the “Savvis C[DN] patents attracted many suitors when they were put up for sale.”³¹ Another such document noted the various “bids and expressions of interest” on these assets, stating that “[e]ach bidder had different goals, valuations, and levels of interest in the various assets.”³²

54. Further, in 2006 Mr. Andrew Swart, then employed at Savvis as a Vice President of Engineering, was involved with aspects of the due diligence discussions with various prospective bidders for the sale of Sandpiper assets discussed above. At that time, there were several competitive bids for the Sandpiper assets. Mr. Swart travelled to San Francisco to meet with several potential buyers of the Sandpiper assets as part of those diligence discussions. In these presentations Mr. Swart gave to potential buyers, the Sandpiper patents were highlighted to the potential buyers. Around that time, Mr. Swart understood that Microsoft had made an offer to buy the Sandpiper assets. The Sandpiper assets, as offered for sale and eventually sold, included the Sandpiper patents. Mr. Swart’s recollection is that in the early stages of this process, bids were in the \$20 million range. Ultimately, the sale to Level 3 was for \$135 million (over \$223 million in today’s dollars).

55. At the time, Microsoft was by far the largest Savvis CDN customer. One publicly filed document appears to show Savvis’s “estimated monthly revenue” from its CDN contract with Microsoft as \$1.1 million, with Savvis’s other nine (anonymized) largest CDN customers totaling \$84,100 in monthly revenue.³³ As detailed throughout this Amended Complaint, Microsoft was also clearly evaluating its upcoming anticipated CDN use and exploring the possibility of building its own

³¹ Limelight case, Dkt. 333, pp.2, 7-8.

³² Limelight Case, Dkt. 355, p. 5.

³³ Limelight Case, Dkt. 484-1, p. 6; *see also id.*, p. 2 (“Also note that I separated the IP-related diligence request - I understand the IP list is comprehensive” and requesting “a complete list of IP assets”); *see also* Limelight Case, Dkt. 484-2, p. 4 (appearing to list Savvis’s revenue from Microsoft as over \$22.5 million in 2004, over \$14 million in 2005, and over \$7.8 million in the first two quarters of 2006).

CDN to insource much of this traffic. One October 2008 article reports that “Microsoft is building its own content delivery network (CDN), with Limelight providing software and engineering support.”³⁴ Another report from around that timeframe specifically listed Microsoft as “interested in acquiring a CDN.”³⁵

56. During this time, several applications to the Asserted Patents had been filed and made public: (a) the ’112 patent (filed on March 18, 2008, published on Sept.25, 2008); (b) the ’871 patent (filed on Sept. 30, 2002; published on Apr. 3, 2003); and (c) the ’964 patent (filed on Oct. 27, 2007, published on Sept. 4, 2008). On information and belief, at least the combination of Microsoft’s interest in acquiring a CDN, Savvis’s CDN business being for sale, Microsoft’s bid on those assets that included the Sandpiper patents as a highlighted asset, and Microsoft being Savvis’s largest CDN customer, led Microsoft to gain actual knowledge of Sandpiper’s issued patents and pending applications.

57. Around the time of Level 3’s acquisition, an executive from Level 3 explained that Level 3 “acquired the business primarily for its intellectual property and architecture.”³⁶ 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.³⁷ An executive from

³⁴ <https://www.datacenterknowledge.com/hyperscalers/limelight-reworks-software-for-microsoft-cdn>.

³⁵ <https://www.streamingmedia.com/Articles/ReadArticle.aspx?ArticleID=65394&pageNum=3>.

³⁶ 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>.

³⁷ 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); 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

Microsoft, Microsoft's then-general manager of Global Foundation 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."³⁸

58. 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."³⁹

59. Ultimately, Level 3 purchased the Savvis CDN business and Sandpiper patents, and Microsoft became one of the largest customers of Level 3. On information and belief, Microsoft remained aware of the Sandpiper patents, including at least specifically the '112 patent, the '871 patent, and the '964 patent, because of Microsoft's previous interest in acquiring a CDN, Microsoft having been Savvis's largest customer, Microsoft's bid on the Savvis assets including the Sandpiper patents that were highlighted to potential buyers as part of that process, and Microsoft becoming one

"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>) ("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.").

³⁸ 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/>.

³⁹ Rich Miller, "Microsoft Assumes Savvis Leases for \$200M," Data Center Knowledge, <https://www.datacenterknowledge.com/hyperscalers/microsoft-assumes-savvis-leases-for-200m>.

of Level 3's largest CDN customers, and Microsoft remained aware of Level 3's continued prosecution of new patents in the portfolio while Microsoft remained a customer, including the '517 Patent, the '738 Patent, the '876 Patent, the '899 Patent, and the '066 patent.

60. Over the years, Microsoft has hired several individuals who previously worked at 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. As explained above, these patents were widely understood to be central to the value of Sandpiper CDN and its predecessors, such that each of these employees had actual knowledge of those patents, including the Asserted Patents or at least the applications leading to them. For instance, one current Microsoft executive in the St. Louis area was recently described on Microsoft's website (in a web page cited by Sandpiper in its *Microsoft I* Complaint that Microsoft has since, it appears, removed) as a "Director" who helps partners "scale their Azure business," and Microsoft touted him as "bring[ing] over two decades of experience in engineering leadership and cloud strategy, including 12 years at CenturyLink (formerly Savvis), where he served as Director of Engineering for Managed Services."⁴⁰ 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.

61. Furthermore, Sandpiper has been involved in significant, widely public litigation regarding infringement of its patents, including three of the Asserted Patents against Microsoft. In addition to the pending suit against Microsoft, these lawsuits include:

- *Sandpiper CDN, LLC v. Google LLC*, Case No. 2:24-cv-03951-AB (C.D. Cal. May 10,

⁴⁰ See *Microsoft I*, Dkt. 41 (Second Amended Complaint, Oct. 30, 2025) at ¶ 54; compare [https://web.archive.org/web/20250807131523/https://dco.microsoft.com/\(S\(pzuaqe2d2sj2jxx20sutonh3\)\)/Team.aspx](https://web.archive.org/web/20250807131523/https://dco.microsoft.com/(S(pzuaqe2d2sj2jxx20sutonh3))/Team.aspx) (August 7, 2025 version of the web page with the language quoted in Sandpiper's *Microsoft I* filing); with <https://dco.microsoft.com/Team.aspx> (rerouting, today, to <https://partner.microsoft.com/en-US/DCO> with no such language).

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 '112 and '517 patents 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 '876 patent asserted against Microsoft in this case.
- *Sandpiper CDN, LLC v. Cox Communications, Inc. et al*, Case No. 1:26-cv-03893-ELR (N.D. Ga., July 11, 2026). In July 2026, Sandpiper sued Cox in the Northern District of Georgia, asserting patents within the same portfolio as those asserted in this case, including the '876 Patent asserted against Microsoft in this case.

62. These lawsuits have received significant press coverage, including through press releases and articles from RPX.⁴¹

63. On information and belief, Microsoft is a member of RPX and receives notices and updates from RPX on patent litigations.⁴²

64. On information and belief, as a member of RPX who receives notices and updates on patent litigations, Microsoft received notices discussing the aforementioned lawsuits from Sandpiper.

65. On information and belief, several of these notices specifically named the U.S. Patent numbers of the patents asserted in those cases, some of which are also Asserted Patents in this case,

⁴¹ 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>.

⁴² 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).

and give an overview of the infringement allegations in those cases.⁴³

66. Furthermore, on information and belief, Microsoft at least became aware of the *Comcast* and *Cox* cases through monitoring the public filings in those cases, including the Complaints, claim constructions entered in *Comcast*, and became aware that Comcast ultimately settled the lawsuit with Sandpiper. On information and belief, Microsoft also had knowledge of the settlement agreement and its contents in *Comcast* before the filing of this lawsuit. As such, Microsoft at least had actual pre-suit knowledge of the ‘876 patent asserted in *Comcast*, *Cox*, and this case, knew about Sandpiper’s infringement allegations regarding Comcast and Cox’s CDN networks, understood that Comcast settled its case, and appreciated the risk of its own infringement of at least the ‘876 patent based on offering its own CDN services.

67. As such, Microsoft was either aware of these lawsuits, including the patents asserted therein, or willfully blind to them, and never approached Sandpiper about a license prior to this lawsuit being filed.

68. In June 2025, Sandpiper CDN sued Microsoft in the Eastern District of Texas, asserting patents sharing a common owner as those asserted in this case. In October 2025 Sandpiper CDN filed a First Amended Complaint in *Microsoft I* showing the history of these two companies

⁴³ RPX, May 12, 2024 “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> (naming “the six patents (8,595,778; 8,645,517; 8,719,886; 9,021,112; 10,924,573) now in suit against Google”); https://insight.rpxcorp.com/litigation_documents/15685653 (RPX website hosting *Sandpiper v Google* Complaint, listing U.S. patent number of ‘112 Patent asserted in this case and containing Sandpiper CDN’s infringement allegations in that case); <https://insight.rpxcorp.com/litigation/txedce-233870-sandpiper-cdn-v-comcast-cable-communications-llc-d-b-a-xfinity#overview> (RPX website hosting docket for *Sandpiper CDN v Comcast* case, with a clickable “Patents-in-Suit” tab indicating “5 Patents-in-Suit” and a “View Complaint” button); <https://insight.rpxcorp.com/litigation/cacdce-925901-sandpiper-cdn-v-google> (RPX website hosting docket for *Sandpiper v. Google* case, with a clickable “Patents-in-Suit” tab indicating “5 Patents-in-Suit” and a “View Complaint” button).

and of Microsoft’s intense interest in acquiring rights to Sandpiper CDN’s patents. Further, a now-public document apparently provided by Savvis in the 2006 timeframe to potential bidders on Savvis’s patents (such as Microsoft), titled “Savvis Content Delivery Related Patents and Published Patent Applications,” specifically listed as a separate item: “ITM Published U.S. Patent Application: US 2003-0065762 A1.”⁴⁴ That published application issued as the ’871 patent asserted in this case. Moreover, the ’964 Patent, the ’738 Patent, and the ’517 Patent each claim priority to the same application and provisional for the ’871 Patent. Thus, even if Microsoft were somehow not aware in 2006 of that patent application, its inclusion in the “ITM” family, or its importance to “Content Delivery” networks like the one Microsoft was planning in 2006 and operates now, Microsoft was certainly aware of all of this information in October 2025 when it received Sandpiper’s Second Amended Complaint.

69. Level 3, after paying \$135 million for the patents now owned by Sandpiper CDN, operated its own ITM service, explaining that “Intelligent Traffic Manager (ITM) is a DNS-based, web administered, global load balancing service,” and that Level 3 practices ITM patents in providing its services.⁴⁵

70. But at least by 2017 Microsoft decided to release its own “Intelligent Traffic Manager,” explaining “Azure Traffic Manager is a global load balancer” that “uses the Domain

⁴⁴ See *Microsoft I*, Dkt. 41, p. 12 n.26 (citing *Level 3 Communications, LLC v. Limelight Networks, Inc.*, 09-589, Dkt. 484-2 (D. Del. Feb. 18, 2009). This was the published application for patent application 10/259,497, and was granted in 2010 as U.S. Patent No. 7,822,871.

⁴⁵ See Intelligent Traffic Manager (ITM) Global Load Balancing Service <https://assets.lumen.com/is/content/Lumen/Service-Guide-ITM-Services-20140423pdf?Creativeid=f6f344ff-d8df-4717-8323-8a2039ac51fa> (“CenturyLink provides ITM Service using the ITM infrastructure of Level 3 Communications.”); see also Level 3, “Legal,” <https://web.archive.org/web/20141012113846/http://www.level3.com/en/legal/> (“Certain Level 3 content delivery services are provisioned using systems, methods, and/or functionality covered by one or more of the following U.S. Patents: ... 7,822,871”).

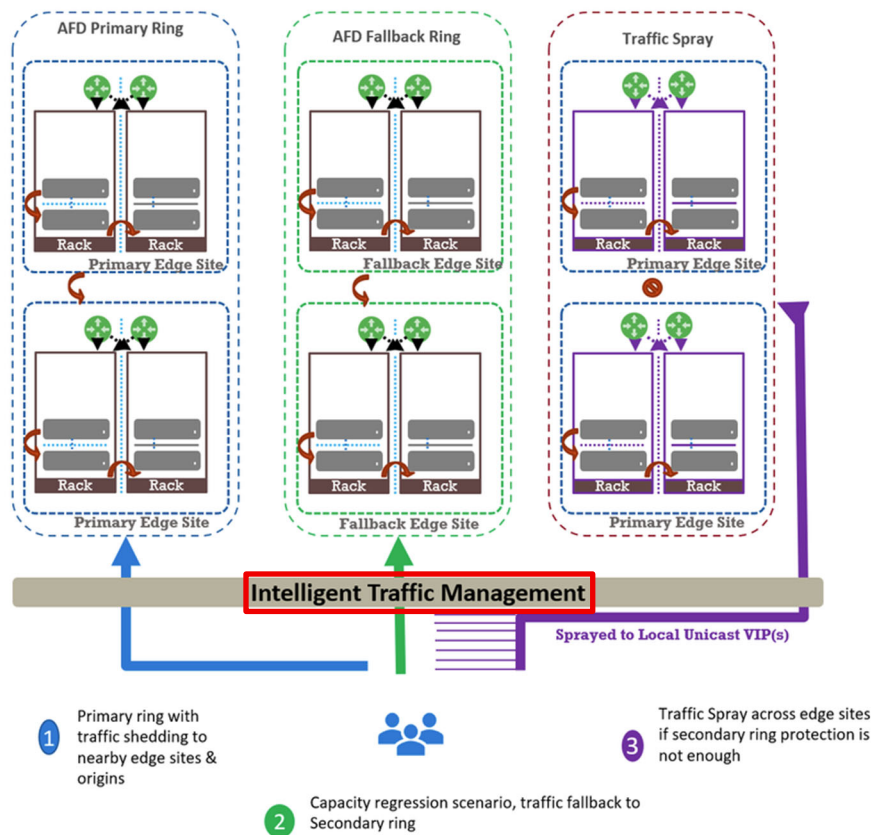
Name System (DNS) protocol.”⁴⁶

71. To this day, Microsoft still refers to its Azure Traffic Manager as an “Intelligent Traffic Manager” in public-facing documents. For example, in response to a CDN outage Microsoft faced just a few months ago, it described: “We have a highly distributed resilient architecture, which protects against failures at the server, rack, site and even at the regional level. This resiliency is achieved by the use of our *intelligent traffic management layer* which monitors failures and load balances traffic at server, rack or edge sites level within the primary ring, supplemented by a secondary-fallback ring which accepts traffic in case of primary traffic overflow or broad regional failures.”⁴⁷ Microsoft further graphically depicted its ITM system in this document.⁴⁸

⁴⁶ See “Microsoft CEO Mentions Containers and Serverless Computing in Keynote,” Sept. 25, 2017, <https://www.sdxcentral.com/news/microsoft-ceo-mentions-containers-and-serverless-computing-in-keynote/> (“As for serverless computing, Microsoft last month added an intelligent traffic manager to its plans with the unveiling of its Azure Event Grid routing service.”); “Architecture best practices for Azure Traffic Manager,” <https://learn.microsoft.com/en-us/azure/well-architected/service-guides/azure-traffic-manager> (“Azure Traffic Manager is a global load balancer that can distribute traffic across multiple Azure regions, zones within a region, or datacenters within those zones. It uses the Domain Name System (DNS) protocol to establish a communication path between a client and your workload’s endpoints.”).

⁴⁷ See “Azure Front Door: Implementing lessons learned following October outages” Dec. 18, 2025 <https://techcommunity.microsoft.com/blog/azurenetworkingblog/azure-front-door-implementing-lessons-learned-following-october-outages/4479416>.

⁴⁸ See *id.*



72. Microsoft was also aware of the '871 patent at least as of 2008 because Microsoft cited the published application of the '871 patent during the prosecution of Microsoft's own patent: U.S. Patent No. 7,467,203, which issued in December 2008. Microsoft also cited that published application in prosecuting Microsoft's U.S. Patent no. 8,236,980 (issued in December 2012) and 8,577,892 (issued in November 2013), and 10,127,295 (issued in Nov. 2018), and 10,855,596 (issued in Dec. 2020, and titled "Load Balancing Among Multiple Endpoint Computing Systems of a Domain").

73. Furthermore, in *Microsoft I*, Microsoft moved to dismiss Sandpiper CDN's allegations of willful patent infringement but the Court denied the motion, noting, among other things, that "the Court finds plausible Plaintiff's pleadings that Defendant intended to infringe." *Microsoft I*, Dkt. 56, p. 5, adopted at Dkt. 104.

74. As demonstrated by the above, and on information and belief, in the 2006 timeframe

Microsoft was looking to build its own CDN to transition away from the services it received from Savvis, and would later receive from Level 3, and knew it needed to own, or obtain a license to, the Sandpiper patents in order to avoid patent infringement. Microsoft thus conducted appropriate diligence on and bid on the Sandpiper assets, including the Sandpiper patents highlighted in those offers (and specifically including two then-existing patent applications that were granted as patents asserted in this case) but ultimately was unwilling to pay what was necessary to avoid infringement, and decided to willfully infringe the Sandpiper patents anyway. In the years after 2006, as Microsoft continued to purchase CDN services from Level 3 while building out its own CDN, Microsoft still never obtained a license to any Sandpiper patents, even though Microsoft continued to receive more information about the granted Sandpiper patents asserted in this lawsuit, and demonstrated its knowledge of these patents at least by its own patent prosecution efforts.

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

SANDPIPER CDN

76. Named after the company that originally pioneered and developed CDN technologies in the 1990s, Sandpiper CDN brings this suit to address Microsoft's ongoing and continued infringement of the patented technology claimed by the Asserted Patents.

77. 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

78. U.S. Patent No. 9,021,112 ("the '112 Patent") is entitled "Content Request Routing and Load Balancing for Content Distribution Networks," and it claims priority to U.S. Patent Application No. 09/982,721, filed on October 18, 2001. *See Ex. A.*

79. 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.⁴⁹ When a server with website content, such as an edge server in a content delivery network, receives multiple requests for website content, delivery of that content can be slow.

80. The inventions claimed by the '112 Patent provide concrete solutions to technical problems facing computer networks. In certain embodiments of the '112 Patent, methods are provided for directing Internet users to the geographically nearest content delivery node using anycast addressing applied specifically and deliberately at the DNS tier, not the content delivery tier.⁵⁰ The '112 Patent claims are directed to a technical solution to a problem that existed in CDN architectures prior to the invention: anycast addressing could not be applied directly to content servers because doing so caused TCP session fragmentation, as packets within a single session could be routed to different physical machines sharing the same anycast address.⁵¹ The '112 Patent's inventors recognized that anycast addressing was uniquely compatible with DNS resolution which uses stateless UDP packets, and built a specific two-tier DNS resolution architecture around that compatibility. The patent claims concrete methods in which CDN DNS servers share a common anycast address, use BGP to advertise that address to the surrounding network, and in some embodiments cause the BGP routing infrastructure itself to perform geographic selection by routing each DNS resolution request to the nearest CDN DNS node based on shortest-path metrics.

81. Additionally, in one specific embodiment, the '112 patent provides a specific two-phase resolution sequence with defined roles for each participant: first, an ISP DNS server resolves

⁴⁹ '112 Patent, 1:36-41.

⁵⁰ '112 Patent, 1:63-2:8.

⁵¹ '112 Patent at 8:27-45.

the subscriber's hostname to the CDN's shared anycast address, a process that does not require any modification of the ISP's infrastructure; and second, one of the CDN DNS servers, selected by the BGP routing fabric based on network proximity, resolves that hostname to the IP address of the content server at its specific node.⁵²

82. According to one claimed solution in the '112 Patent, the CDN DNS server is associated with a respective plurality of content servers at its node, and requests for content are resolved to the DNS server, and then resolved to an IP address for one of the associated content servers.⁵³ That route advertisement to the DNS server can be used as a dynamic load-balancing and failover mechanism: when a node becomes overloaded, the advertisement is withdrawn, automatically redirecting subsequent DNS resolution requests to the another node.⁵⁴ These are concrete, specific implementations rooted in the technical realities of BGP routing, UDP-based DNS, and the incompatibility of anycast with TCP sessions, and not a generic instruction to use a content delivery network for a desirable result.

83. U.S. Patent No. 7,822,871 (the '871 Patent) is entitled "Configurable Adaptive Global Traffic Control and Management" and it claims priority to U.S. Patent Application No. 10/259,497, filed on September 30, 2002, which is a continuation of U.S. Patent Application No. 60/325,177, filed on September 28, 2001. *See Ex. B.*

84. The inventions claimed by the '871 Patent provide concrete solutions to these technical problems facing computer networks. In one embodiment, the '871 Patent provides a means to distribute network traffic according to a configurable set of rules that take into account key factors such as server availability, specific requirements of content providers deploying the invention,

⁵² '112 Patent at 8:49-65.

⁵³ '112 Patent at 9:3-21.

⁵⁴ '112 Patent at 9:3-21.

including distribution based upon geography, position in IP address space, and load share, and the state of the network at any given moment, including measures of network latency.⁵⁵

85. For example, in one claimed embodiment of the '871 Patent, there are three distinct, concrete steps of (1) maintaining a traffic distribution policy used by a domain name server in an adaptive traffic control (ATC) name server network to determine the IP address of a server in response to a request from a client to resolve a hostname, (2) monitoring the operational status of the ATC name server network through a specific event-processing pipeline, and (3) maintaining and broadcasting those processed operation events to domain name servers in the ATC name server network.⁵⁶ These are specific, concrete operational steps that constitute a technical improvement to how distributed DNS infrastructure is managed, and not a generic instruction to monitor or administer a network.

86. U.S. Patent No. 7,860,964 (the '964 Patent) is entitled "Policy-Based Content Delivery Network Selection," and it is a continuation-in-part claiming priority to U.S. Patent Application No. 10/259,497, filed on September 30, 2002, which is a continuation of U.S. Patent Application No. 60/325,177, filed on September 28, 2001. *See Ex. C.*

87. The '964 Patent relates to network traffic management, and more specifically to configurable, adaptive, global traffic control and management in networks such as the Internet.⁵⁷ As content delivery networks grew in commercial importance, content providers increasingly contracted with multiple CDN operators simultaneously, providing redundancy, cost competition, and geographic coverage. Prior DNS-based traffic management, however, could direct a client to only a single CDN provider at a time, through a statically-configured DNS record pointing to that CDN's

⁵⁵ '871 Patent at 1:43-52.

⁵⁶ '871 Patent, 27:39-63.

⁵⁷ '964 Patent, 1:24-27.

namespace. The invention is described in a scenario in which DNS service is provided by a third party on behalf of a content or applications provider.⁵⁸

88. The inventions claimed by the '964 Patent provide concrete solutions to these technical problems facing computer networks. In one embodiment, the '964 Patent describes a subscriber maintaining an even load across all subscriber servers by setting a load share value, with traffic shedding to an overflow server defined by a CName for a CDN when all servers near capacity.⁵⁹ In another embodiment, the specification describes a concrete multi-CDN configuration in which an ATC is configured to return the domain name of one CDN 1.5 times as often as it returns the domain name of a second CDN, thereby implementing a 1.5-to-1 load share ratio between two distinct CDN providers entirely at the DNS layer.⁶⁰

89. One claimed embodiment of the '964 Patent is directed to a “method operable in a framework wherein resources of a content provider are delivered to clients from one of a plurality of distinct content delivery network (CDN) domains distinct from a content provider domain.” The claim requires “obtaining one or more content provider policies relating to delivery of said resources, at least one of said policies relating to a relative load share for each of said plurality of CDN domains.”⁶¹ The load share policy requirement is technically specific: the policy must quantify the relative fraction of traffic that each distinct CDN domain is to receive, making it a machine-enforceable distribution rule. The claim further requires “selecting, based at least in part on said one or more content provider policies, one of said plurality of CDN domains.”⁶² And when a CDN domain is selected, the claim requires “providing the client with a domain name associated with the selected

⁵⁸ '964 Patent, 3:4-8.

⁵⁹ '964 Patent, 27:8-15.

⁶⁰ '964 Patent 28:7-32.

⁶¹ '964 Patent, Claim 19.

⁶² '964 Patent, Claim 19.

CDN.”⁶³ The requirement to provide a domain name is a technically significant and specific implementation choice: by returning a domain name in the CDN’s own namespace, the method directs the client’s DNS resolver to continue resolution within the selected CDN’s own infrastructure, enabling that CDN to apply its own geographic and load-based edge selection for the final resolution step.

90. U.S. Patent No. 9,660,876 (the ’876 Patent) is entitled “Collector Mechanism In Content Delivery Network” and it claims priority to U.S. Application No. 13/802,440, filed March 13, 2013, which in turn claims priority from U.S. Provisional Application No. 61/737,072, filed December 13, 2012. *See Ex. D.*

91. As CDN technology developed, the type of the content served and delivered by CDNs evolved. As a result, CDNs—originally configured to treat requests for all content the same—experienced issues as they started receiving different requests for different types of content, such as video, images, written work, or other multimedia content. Further compounding the issue was the challenge in configuring CDNs to account for different types of parameters associated with delivering content, including responsibilities, security, quality, and the like.

92. The inventors of the ’876 Patent identified these technical problems and invented specific and technical solutions to address them. In particular, the inventors developed, *inter alia*, a specific and technical solution whereby “state data are used to inform a peering policy of a set of caches,” where a collector system “produce[s] state data relating to and based on information represented in said event data of said multiple event streams.”⁶⁴ The inventors also discovered that using the state data relating to the event streams to inform a peering policy empowered the customization of computing components in a CDN, which improves computational efficiency of the

⁶³ ’964 Patent, Claim 19.

⁶⁴ ’876 Patent, claim 1.

CDN through specialization. Moreover, the disclosed inventions of the as the inventors explained to the Patent Office, '876 Patent are not abstract ideas, but are “applicable and limited by the claims to content delivery networks and provide[] a concrete way to support efficient and scalable content delivery in such networks . . . [and] “[are] necessarily rooted in computer technology in order to overcome a problem specifically arising in the realm of computer networks.”⁶⁵ The technical solution of the '876 Patent provides CDNs with the infrastructure and logic to handle specific requests for specific users in a more computationally efficient manner that was more scalable than other existing systems.

93. Indeed, the claims of the '876 Patent include specific and technical solutions, using specific structures, that were not conventional at the time, and the '876 Patent describes how the solutions are implemented, as recited in the method of independent claim 1. First, using the claimed state data relating to the event streams to inform a peering policy allows for the customization of computing components in a CDN to improve computational efficiency of the CDN through specialization. Second, using the claimed state data to inform a peering policy for a set of peer caches improves fault tolerance within a CDN to reduce outages. Third, certain CDNs implementing embodiments claimed by the are able to account for the ordering receiving multiple event '876 Patent of the event streams to improve efficiency in configuring the CDNs by streams, each comprising a timestamp for the event.

94. U.S. Patent No. 9,647,899 (the '899 Patent) is entitled “Framework Supporting Content Delivery with Content Delivery Services,” and it was filed December 14, 2012, claiming priority to U.S. Provisional Application No. 61/737,072, filed December 13, 2012. *See Ex. E.*

⁶⁵ *See Applicant Arguments/Remarks Made in an Amendment* dated June 2, 2016, p. 12 (successfully traversing a Patent Office rejection under 35 U.S.C. § 101 to establish that the patent application for the '876 patent is directed to patent eligible subject matter) (citing *DDR Holdings, LLC v. Hotels.Com, L.P.*, 773 F.3d 1245, 1257 (Fed. Cir. 2014)).

95. Content delivery networks are composed of multiple distinct distributed services operating concurrently, for example collector services for monitoring, analytics, and popularity; reducer services for logging, monitoring, and alarming; control services for control resource distribution and localized feedback; configuration services for configuration validation and resource generation; and primary delivery services including rendezvous and caching.⁶⁶ A technical problem in CDN design was that these services operated in relative isolation, where each generated its own operational events and maintained its own state, but there was no systematic mechanism by which the operational event data produced by one service would flow as input into other services, be aggregated into CDN-wide state information, and be used to automatically reconfigure how those services were interconnected.

96. The inventions claimed by the '899 Patent provide concrete solutions to this problem through a specific framework in which event output, state, and service configuration information are systematically connected. As part of the operation of each service or kind of service, a service may produce event information corresponding to events relating to that service, and may obtain state information from other CDN services or components and may generate state information for use by other CDN services or components.⁶⁷ In one embodiment, a minimal CD service is both an event source and a control sink, meaning that all CDN services consume control information and generate events.⁶⁸ This mutual interconnection enables a specific technical improvement: a closed feedback loop where each service generates events and receives control, and the resulting events and state flow continuously through the network.

97. Claim 1 requires that some CD services in the CDN “produce event output relating

⁶⁶ '899 Patent at col. 11:52-12:9.

⁶⁷ '899 Patent at col. 8:18-49.

⁶⁸ '899 Patent, 9:41-53.

to their respective operation, wherein at least some of said event output from at least some of said CD services is provided to at least some of said CD services as event input.”⁶⁹ This cross-service event flow is the foundation of a claimed feedback architecture. Claim 1 also requires specific technical processes based on those interconnected event outputs, requiring the services to “maintain service configuration information specifying associations between CD service instances,” “obtain state information from said CD services,” and “modify said service configuration information to produce modified service configuration information based at least on the state information.”⁷⁰ The “service configuration information specifying associations between CD service instances” is a technically specific data structure and the claim requires that this configuration be modified based on the state information derived from event flows.

98. U.S. Patent No. 10,116,738 (the '738 Patent) is entitled “Detecting anomalous conditions in a name server network,” and it was filed November 30, 2015, claiming priority to U.S. Provisional Application No. 60/325,177, filed September 28, 2001. *See Ex. F.*

99. The '738 patent is generally directed to network traffic management and configurable traffic control and management. As the volume of Internet traffic grows, content and application providers increasingly need to deliver content from multiple servers at widely separated locations to sustain a good end-user experience under high traffic loads. This creates several challenges, including guaranteeing fault tolerance in the face of individual server or name server failures and maintaining high performance as network conditions change, particularly since the state of the network can shift at any given moment.

100. The inventions claimed by the '738 Patent provide concrete solutions to addresses these challenges with a system that lets a subscriber define, through a web-based graphical user

⁶⁹ '899 Patent, claim 1.

⁷⁰ '899 Patent, claim 1.

interface, policies governing how service requests are distributed among the subscriber's servers, while a network of name servers responsible for resolving hostnames to those servers is continuously monitored for its own operational health.

101. Claim 1 requires receiving policies for distribution of service requests to one or more servers in a subscriber server network. The method further requires monitoring the operational status of name servers in a name server network that provides hostname resolution services for the servers in the subscriber server network. When monitoring detects an error or anomalous condition at a particular name server, the system automatically creates a policy reflecting that name server's status, so that subsequent hostname resolution decisions across the name server network take the condition into account. The subscriber is also proactively notified of the detected condition through the same web-based graphical user interface used to configure policies, closing the loop between real-time network monitoring, adaptive policy management, and subscriber visibility. This combination allows the traffic-management system to respond dynamically to name server failures or anomalies without manual intervention, while keeping the subscriber informed so they can take further action if needed.

102. U.S. Patent No. 8,156,066 (the '066 Patent) titled "Rule-Based Content Request Handling," issued on April 10, 2012, from an application claiming priority to an April 9, 2008 provisional application. *See Ex. G.*

103. The inventions claimed in the '066 Patent address a specific technical problem facing content delivery networks: how a shared distribution infrastructure serving many independent content providers could apply each provider's own, independently configured access and handling rules to incoming content requests, at scale, without requiring every provider to build and operate separate request-handling infrastructure. The '066 Patent's solution involves deriving attribute values not directly supplied by the requester from attribute values already present in an incoming content request, such as deriving the requester's likely geographic region from the

requester's Internet Protocol address. Both the originally supplied and the newly derived attribute values are evaluated against a hierarchy of decision nodes, arranged so that a request is resolved only after propagating through different levels of specific logic culminating in a single decision that dictates whether the request is allowed, denied, or redirected.

104. Claim 14 of the '066 Patent recite an architecture including: (a) receiving a content request that includes a first set of attribute values; (b) using at least one attribute value from that first set to determine a second, distinct set of attribute values; (c) traversing a hierarchy of decision nodes, each implementing logic based on an attribute value from either set; and (d) yielding, from the last node reached in that hierarchy, a decision dictating how the request is to be answered. Claim 14 requires that the request-disposition decision be reached only through a specific implementation mechanism: deriving a second, distinct set of attribute values from the request's original attributes and then routing evaluation through a defined multi-node hierarchy that connects its logic to one of those two attribute generations. This defines specific technical implementation and does not simply recite a desired result of deciding whether to allow, deny, or route a request based on the requester's characteristics. Claim 14's steps thus describe a specific data-transformation and control-flow architecture that a computer must implement to reach the claimed decision, not merely a functional goal that architecture achieves.

105. U.S. Patent No. 8,645,517 (the "'517 Patent") is entitled "Policy-Based Content Delivery Network Selection." The '517 Patent claims priority to U.S. Patent Application Number 10/259,497, filed on September 20, 2002. *See Ex. H.*

106. When using CDN technology, challenges arise related to network traffic associated with requests for content. For example, companies that want to provide their content to end users using a CDN face issues regarding directing network traffic to serve end users' requests. As one example, issues arise when trying to deliver content from multiple servers at geographically-separated

location with suitable end-user experiences under high traffic loads.

107. The inventors of the '517 Patent understood these challenges faced by CDN providers and their customers relating to directing network traffic and end-user requests for content. The inventors devised solutions utilizing Domain name Server (DNS) technology and allowing for the resolution of requests to multiple CDNs based on various policies. Embodiments of the disclosed invention solve network traffic issues such as server failure, and controlling the distribution of requests according to economic or contractual parameters, by providing a network distribution infrastructure that can be configured with network traffic rules. These rules can account for factors like server availability, geolocation, load, and latency. The '517 Patent addresses specific needs in the art via specific combinations, which were not conventional at the time of the invention of the '517 Patent, and the '517 Patent describes how specific technical solutions are achieved. For instance, content providers must address how to handle requests from end users, in some cases in accordance with policy constraints faced by the content providers, even as network conditions affecting the Internet are subject to change. The inventors of the '517 Patent developed methods to address these issues, including methods in which a server network implements policy-based traffic direction. For example, embodiments of the invention in the '517 Patent provide a specific graphical user interface for obtaining at least one policy for the direction of network traffic, such as using a decision tree with resource and branch nodes. The various criteria that can be used may relate to IP addresses or geographic zones, with answers consisting of IP addresses and CNAMEs, for example.

THE ACCUSED MICROSOFT FUNCTIONALITIES

108. Microsoft's Azure is a cloud computing platform offering a range of cloud services for businesses and developers.⁷¹ Core service categories include cloud computing, cloud storage, databases, networking, application hosting and development, and security. Microsoft additionally

⁷¹ <https://azure.microsoft.com/en-us/resources/cloud-computing-dictionary/what-is-azure/>

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.⁷²

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>

109. Microsoft's Azure CDN products offer global coverage through distributed points of presence (PoPs) including throughout North America.⁷³ Microsoft's Azure CDN products utilize repeater servers (e.g., EdgeNodes or Endpoints) for replicating (e.g., caching) content from origin servers.⁷⁴ 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.⁷⁵ Azure CDN was announced May 7, 2018 as a public preview.⁷⁶

110. Azure Front Door is Microsoft's "modern" CDN product featuring similar content replication and delivery functionality with added scalability and performance.⁷⁷ Azure Front Door

⁷² <https://learn.microsoft.com/en-us/azure/cdn/cdn-overview>; <https://learn.microsoft.com/en-us/azure/frontdoor/front-door-overview>.

⁷³ <https://azure.microsoft.com/en-us/products/cdn>; <https://learn.microsoft.com/en-us/azure/cdn/cdn-pop-locations>.

⁷⁴ <https://learn.microsoft.com/en-us/azure/cdn/cdn-overview>

⁷⁵ <https://learn.microsoft.com/en-us/azure/cdn/cdn-features>

⁷⁶ <https://azure.microsoft.com/en-us/blog/announcing-microsoft-s-own-cdn-network/>

⁷⁷ <https://learn.microsoft.com/en-us/azure/frontdoor/front-door-overview>

provides autoscaling and global traffic distribution with load balancing to ensure high availability.⁷⁸

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>

111. Microsoft further offers Azure Policy, enabling organizations to create, assign, and manage policies to enforce compliance and security across their cloud resources.⁷⁹ 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.⁸⁰

112. Microsoft further offers Azure Resource Manager, providing infrastructure management and access control functionality in a unified management platform.⁸¹ Azure Resource Manager was introduced in 2014.⁸²

113. Microsoft further offers Azure Traffic Manager, a DNS-based traffic load balancer that distributes traffic to public-facing applications across global Azure regions, providing public endpoints with high availability and quick responsiveness.⁸³ Traffic Manager uses DNS to direct clients to specific service endpoints based on configured routing methods and provides distribution

⁷⁸ <https://learn.microsoft.com/en-us/azure/frontdoor/front-door-faq>

⁷⁹ <https://learn.microsoft.com/en-us/azure/governance/policy/overview>

⁸⁰ <https://sonraisecurity.com/blog/what-is-azure-policy-all-you-need-to-know/>

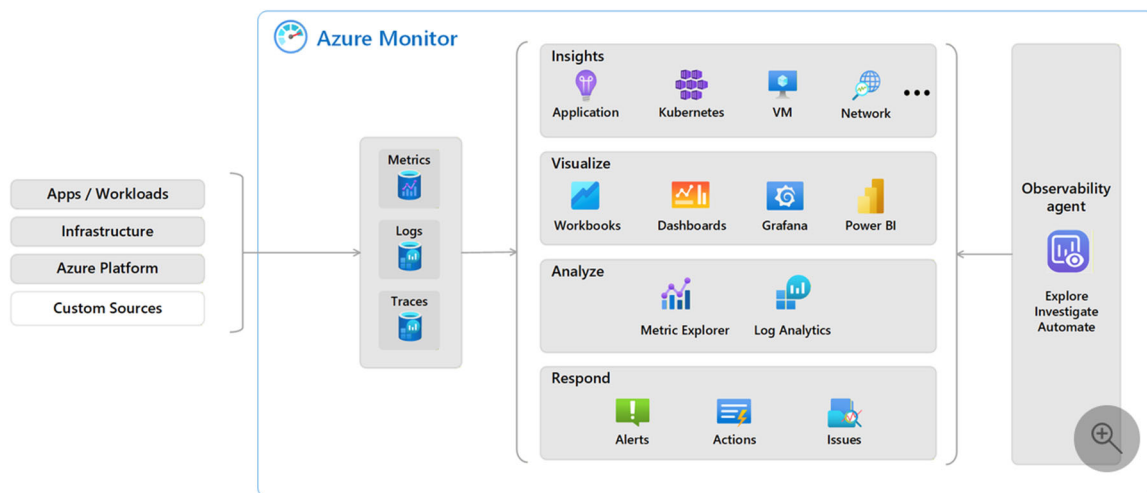
⁸¹ <https://learn.microsoft.com/en-us/azure/azure-resource-manager/management/overview>

⁸² <https://web.archive.org/web/20161029053752/https://azure.microsoft.com/en-us/documentation/articles/resource-manager-deployment-model/>

⁸³ <https://learn.microsoft.com/en-us/azure/traffic-manager/traffic-manager-overview>

of traffic according to one of several routing methods, as well as continuous monitoring of endpoint health with automatic failover when endpoints fail. Traffic Manager caches DNS responses through configurable time-to-live (TTL) values which are configurable.⁸⁴

114. Microsoft further offers Azure Monitor, a unified observability service for collecting, analyzing, and acting on telemetry data from cloud and hybrid environments, enabling users to understand the health, performance, and reliability of Azure applications and infrastructure resources by bringing together metrics, logs, traces, and events into a single user experience.⁸⁵



<https://learn.microsoft.com/en-us/azure/azure-monitor/fundamentals/overview>

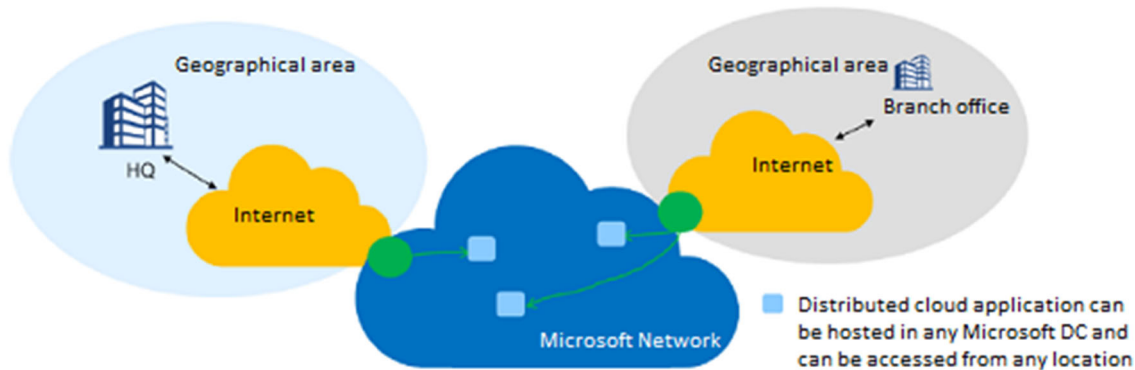
Azure Monitor collects logs and metrics from all resources across Azure services and integrates this monitoring into the Azure portal for each service. Azure Monitor also proactively notifies subscribers when issues are identified in collected data.

115. Microsoft further offers Azure Peering Service, a networking service that enhances connectivity to Microsoft cloud services such as Microsoft 365, Dynamics 365, SaaS services, Azure, and others. Through Azure Peering service, Microsoft partners with internet service providers (“ISPs”) to provide reliable and high-performing public connectivity through optimal routing from

⁸⁴ <https://learn.microsoft.com/en-us/azure/traffic-manager/traffic-manager-overview>

⁸⁵ <https://learn.microsoft.com/en-us/azure/azure-monitor/fundamentals/overview>

and to the Microsoft network.⁸⁶



<https://learn.microsoft.com/en-us/azure/peering-service/about>

Through Azure Peering Service, Microsoft ensures traffic stays on its own high-capacity, low-latency, and high-reliability Microsoft global network until traffic is as close to its destination as possible. Azure Peering Service further provides geo-redundancy by interconnecting Microsoft’s network with service providers at multiple metro locations, such that if one Azure Edge has degraded performance, traffic routes via alternative sites.⁸⁷

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

117. The aforementioned products and services are collectively the “Accused Microsoft Functionalities.”

⁸⁶ <https://learn.microsoft.com/en-us/azure/peering-service/about>

⁸⁷ <https://learn.microsoft.com/en-us/azure/peering-service/about>

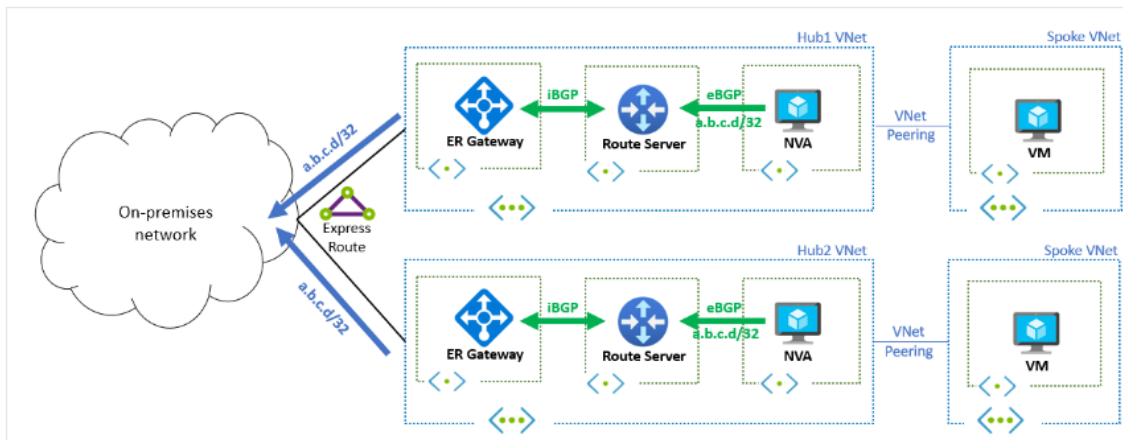
COUNT I: INFRINGEMENT OF THE '112 PATENT

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

119. Microsoft directly infringes, either individually or jointly through the direction and control of others, at least claim 1 of the '112 Patent through one or more of the Accused Microsoft Functionalities. For instance, Microsoft Front Door CDN comprises a network through which Microsoft provides a plurality of Domain Name System servers associated with a Content Delivery Network that share a common anycast address, and where each CDN DNS server is associated with a respective plurality of content servers.

Topology

In the design of this scenario, the same IP address is advertised from virtual networks in different Azure regions, where network virtual appliances (NVAs) advertise the application's IP address through Azure Route Server. The following diagram depicts two simple hub and spoke topologies, each in a different Azure region. An NVA in each region advertises the same route (`a.b.c.d/32` in this example) to its local Azure Route Server (the route prefix must not overlap with Azure and on-premises networks). The routes are further propagated to the on-premises network through ExpressRoute. When application users want to access the application from on-premises, the DNS infrastructure (not covered by this document) resolves the DNS name of the application to the anycast IP address (`a.b.c.d`), which the on-premises network devices route to one of the two regions.



<https://learn.microsoft.com/en-us/azure/route-server/anycast>

120. Microsoft causes said plurality of CDN DNS servers to be authoritative for a hostname associated with a content provider by causing said common anycast address to be associated with said hostname.

Resolution and delegation

There are two types of DNS servers:

- An *authoritative* DNS server hosts DNS zones. It answers DNS queries for records in those zones only.
- A *recursive* DNS server doesn't host DNS zones. It answers all DNS queries by calling authoritative DNS servers to gather the data it needs.

Azure DNS provides an authoritative DNS service. It doesn't provide a public recursive DNS service. For information about Azure's private DNS recursive service, see [What is Azure DNS Private Resolver](#). By default, cloud services and VMs in Azure are configured to use a recursive DNS service that is provided separately as part of Azure's infrastructure. For information on how to change these DNS settings, see [Name Resolution in Azure](#).

DNS clients in PCs or mobile devices typically call a recursive DNS server to do any DNS queries the client applications need.

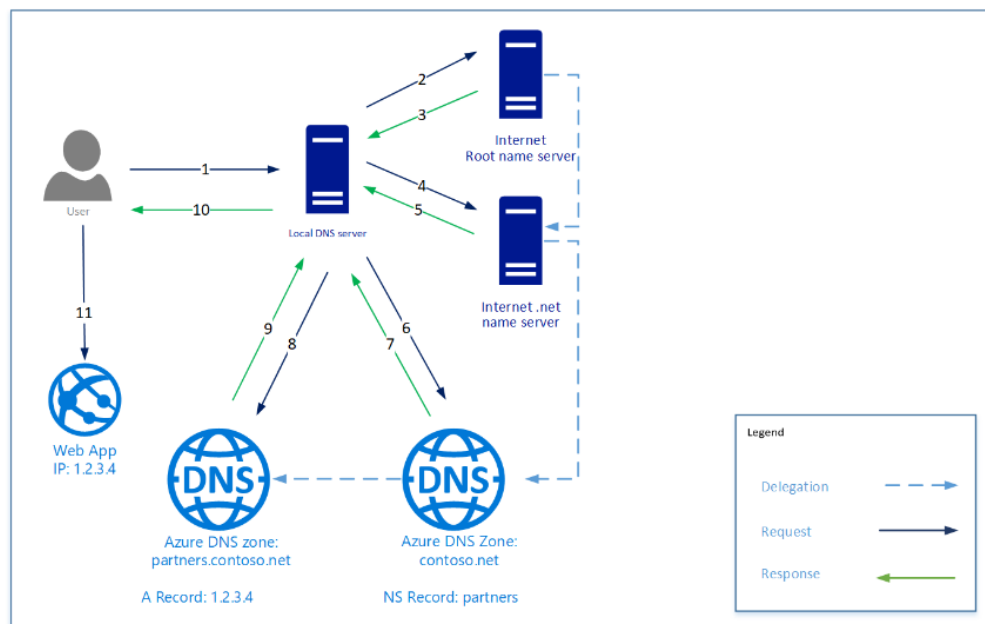
When a recursive DNS server receives a query for a DNS record such as `www.contoso.com`, it first needs to find the authoritative name server hosting the `contoso.com` domain. To find the name server, it starts at the root name servers, and from there finds the name servers hosting the `com` domain. It then queries the `com` name servers to find the name servers hosting the `contoso.com` domain. Finally, it's able to query these name servers for `www.contoso.com`.

This procedure is called resolving the DNS name. Strictly speaking, DNS resolution includes more steps such as following CNAMEs, but that's not important to understanding how DNS delegation works.

How does a parent zone point to the name servers for a child zone? It does this using a special type of DNS record called an NS record (NS stands for 'name server'). For example, the root zone contains NS records for `com` and shows the name servers for the `com` domain. In turn, the `com` domain contains NS records for `contoso.com` that display name servers for the `contoso.com` domain. Setting up the NS records for a child zone in a parent zone is called delegating the domain.

<https://learn.microsoft.com/en-us/azure/dns/dns-domain-delegation>

The following image shows an example DNS query. The `contoso.net` and `partners.contoso.net` zones are public DNS domains that have been delegated to Azure's authoritative DNS servers.



<https://learn.microsoft.com/en-us/azure/dns/dns-domain-delegation>

13.107.236.4 [Copy] [Star] [Anycast] [Hosting] [Nameserver]

Summary

Location	Redmond, Washington, US
ASN	AS8075 — Microsoft Corporation
Hostname	ns1-04.azure-dns.com
Range	13.104.0.0/14
Company	Microsoft Corporation
Hosted domains	8

Anycast **true** Anycast instances identified in 26 countries and 38 cities

AS type **Hosting**

Abuse contact abuse@microsoft.com [Copy]

https://ipinfo.io/13.107.236.4?lookup_source=search-bar

121. Microsoft, responsive to a request for content associated with the content provider and issued by a client, said request including at least said hostname, causes said hostname to be resolved to said common anycast address by an Internet Service Provider DNS server, and then by one of said plurality of CDN DNS servers, resolves said hostname to identify an IP address for use by the client to retrieve the content from a content server. For example, Microsoft causes an ISP DNS server to resolve office.com to the anycast IP address of Azure DNS servers at 150.171.21.5, which then in turn resolve a hostname to an Azure Front Door IP address at 13.107.6.156.

```
$ dig -4 @74.40.74.40 +trace office.com
...
. 18994 IN NS a.root-servers.net.
...
office.com. 172800 IN NS ns1-05.azure-dns.com.
office.com. 172800 IN NS ns2-05.azure-dns.net.
office.com. 172800 IN NS ns3-05.azure-dns.org.
office.com. 172800 IN NS ns4-05.azure-dns.info.
...
office.com. 10 IN A 13.107.6.156
;; Received 55 bytes from 150.171.21.5#53(ns2-05.azure-dns.net) in 14 ms
```

COUNT II: INFRINGEMENT OF THE '871 PATENT

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

123. Microsoft directly infringes, either individually or jointly through the direction and control of others, at least claim 44 of the '871 Patent through operation of one or more of the Accused Microsoft Functionalities. For instance, Microsoft Azure Traffic Manager operates global DNS name servers in connection to Azure Resource Manager, Azure Portal, Azure Monitor, and related control plane management interfaces to perform a method of an adaptive traffic control framework.

What is Azure Traffic Manager?

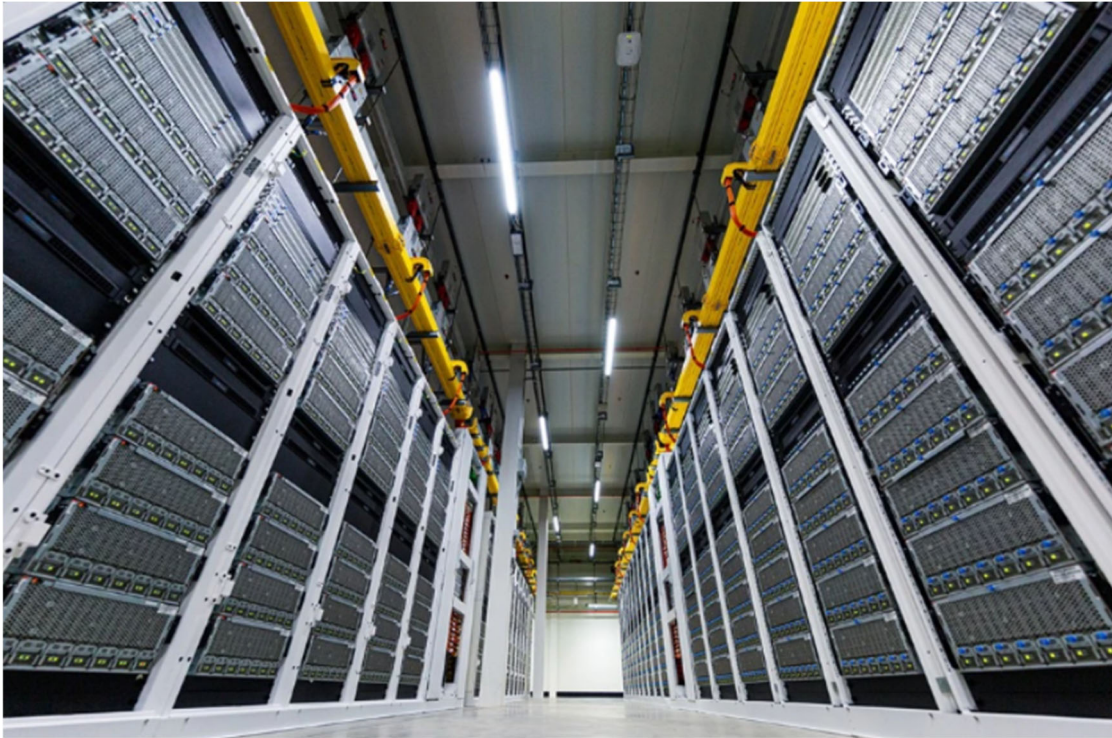
 Summarize this article for me

Azure Traffic Manager is a DNS-based traffic load balancer that distributes traffic to your public-facing applications across global Azure regions. Traffic Manager provides your public endpoints with high availability and quick responsiveness.

Traffic Manager uses DNS to direct client requests to the appropriate service endpoint based on a traffic-routing method. Traffic Manager also provides health monitoring for every endpoint. The endpoint can be any Internet-facing service hosted inside or outside of Azure. Traffic Manager provides a range of traffic-routing methods and endpoint monitoring options to suit different application needs and automatic failover models. Traffic Manager is resilient to failure, including the failure of an entire Azure region.

<https://learn.microsoft.com/en-us/azure/traffic-manager/traffic-manager-overview>

124. Microsoft Azure features a server network comprising hardware in combination with software. For instance, Microsoft cloud datacenters include server hardware with Azure service software operating on this hardware.



<https://azure.microsoft.com/en-us/blog/the-anatomy-of-a-datacenter-how-microsofts-datacenter-hardware-powers-the-microsoft-cloud/>

125. Microsoft further maintains at least one policy used by at least one domain name server in an adaptive traffic control (ATC) name server network to determine the IP address of a server in a server network in response to a request from a client to resolve a hostname. For instance, Traffic Manager name servers for trafficmanager.net receive DNS requests and return endpoints based on configured endpoint state, health, and routing method.

How Traffic Manager Works

✦ Summarize this article for me

Azure Traffic Manager enables you to control the distribution of traffic across your application endpoints. An endpoint is any Internet-facing service hosted inside or outside of Azure.

Traffic Manager provides two key benefits:

- Distribution of traffic according to one of several [traffic-routing methods](#)
- [Continuous monitoring of endpoint health](#) and automatic failover when endpoints fail

When a client attempts to connect to a service, it must first resolve the DNS name of the service to an IP address. The client then connects to that IP address to access the service.

The most important point to understand is that Traffic Manager works at the DNS level which is at the Application layer (Layer-7). Traffic Manager uses DNS to direct clients to specific service endpoints based on the rules of the traffic-routing method. Clients connect to the selected endpoint **directly**. Traffic Manager isn't a proxy or a gateway. Traffic Manager doesn't see the traffic passing between the client and the service.

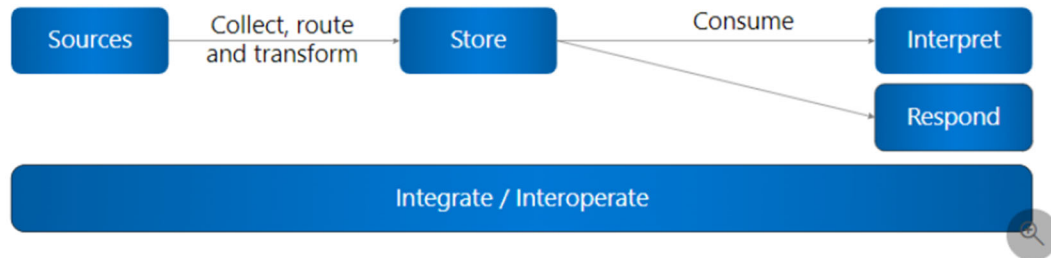
Traffic Manager uses profiles to control traffic to your cloud services or website endpoints. For more information about profiles, see [Manage an Azure Traffic Manager profile](#).

<https://learn.microsoft.com/en-us/azure/traffic-manager/traffic-manager-how-it-works>

126. Microsoft monitors the operational status of the ATC name server network comprising receiving, by an ATC network monitoring mechanism in the ATC administrative framework, events sent from at least one monitoring mechanism of the at least one domain name server. For instance, Azure Monitor monitors operational status of name servers in the network.

Azure Monitor is a comprehensive monitoring solution for collecting, analyzing, and responding to monitoring data from your cloud and on-premises environments. You can use Azure Monitor to maximize the availability and performance of your applications and services. It helps you understand how your applications are performing and allows you to manually and programmatically respond to system events.

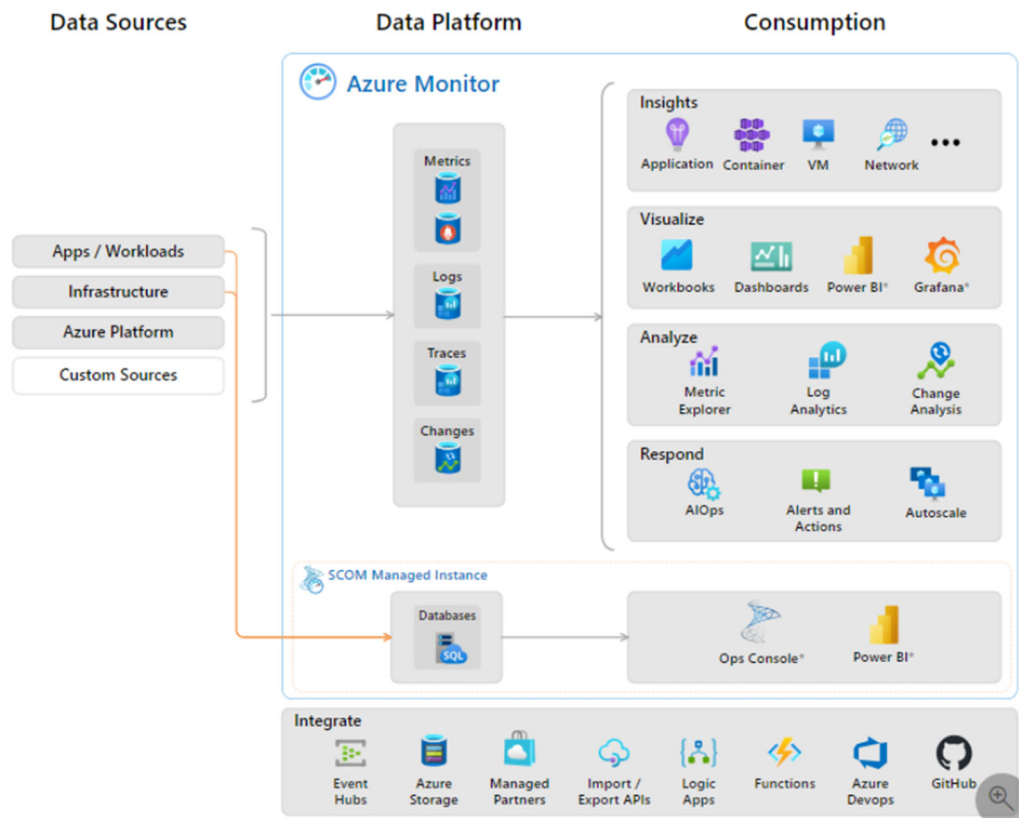
Azure Monitor collects and aggregates the data from every layer and component of your system across multiple Azure and non-Azure subscriptions and tenants. It stores it in a common data platform for consumption by a common set of tools that can correlate, analyze, visualize, and/or respond to the data. You can also integrate other Microsoft and non-Microsoft tools.



The diagram above shows an abstracted view of the monitoring process. A more detailed breakdown of the Azure Monitor architecture is shown in the [High level architecture](#) section below.

<https://learn.microsoft.com/en-us/azure/azure-monitor/overview>

The following diagram shows a high-level architecture view of Azure Monitor.



<https://learn.microsoft.com/en-us/azure/azure-monitor/overview>

Monitor Azure DNS

Article • 01/30/2025 • 2 contributors

[Feedback](#)

In this article

[Collect data with Azure Monitor](#)

[Use Azure Monitor tools to analyze the data](#)

[Export Azure Monitor data](#)

[Use Kusto queries to analyze log data](#)

[Show 2 more](#)

Azure Monitor collects and aggregates metrics and logs from your system to monitor availability, performance, and resilience, and notify you of issues affecting your system. You can use the Azure portal, PowerShell, Azure CLI, REST API, or client libraries to set up and view monitoring data.

Different metrics and logs are available for different resource types. This article describes the types of monitoring data you can collect for this service and ways to analyze that data.

<https://learn.microsoft.com/en-us/azure/dns/monitor-dns>

127. As an additional example, Microsoft monitors the operational status of its DNS servers, for instance through the Azure Service Health service.

Service Health notifications



Summarize this article for me

In this article

[View Service Health notifications](#)

[Service Health notification types](#)

[Related content](#)

Azure Service Health notifications are system-generated alerts that inform you about Azure service problems or events that affect your resources. The subscription's [Azure activity log](#) records these notifications as part of logging many events in Azure. The Azure portal then displays them under [Azure Service Health](#).

When Azure needs to communicate something about service health, such as an outage, upcoming maintenance, or an account-specific alert, it creates a Service Health event in your activity log.

<https://learn.microsoft.com/en-us/azure/service-health/service-health-notifications-properties>

128. Microsoft consolidates the events to form consolidated events, and processes the consolidated events. For instance, Azure Monitor collects activity logs, resource logs, metrics, and other telemetry which can be routed to a Log Analytics workspace to be queried, correlated, aggregated, filtered, and transformed.

Activity log in Azure Monitor



Summarize this article for me

Azure Monitor records management operations for your Azure resources through the activity log feature. The activity log records operations like creating a virtual machine, changing a key vault access policy, or Resource Manager deployment errors. These management operations are also called *control plane* operations. Use the activity log to review or audit this information, or create an alert to be proactively notified when an event occurs.

In contrast to the activity log, *Azure resource logs* capture *data plane* operations performed within a resource. For example, these operations include getting a secret from a key vault or making a request to a database. Resource logs aren't collected by default and require configuration with a *diagnostic setting*.

<https://learn.microsoft.com/en-us/azure/azure-monitor/platform/activity-log>

129. Further, Microsoft Sentinel allows Windows DNS server logs to be streamed and filtered via the Azure Monitor Agent.

DNS over AMA connector reference - available fields and normalization schema



Summarize this article for me

Microsoft Sentinel allows you to stream and filter events from your Windows Domain Name System (DNS) server logs to the `ASimDnsActivityLog` normalized schema table. This article describes the fields used for filtering the data, and the normalization schema for the Windows DNS server fields.

The Azure Monitor Agent (AMA) and its DNS extension are installed on your Windows Server to upload data from your DNS analytical logs to your Microsoft Sentinel workspace. You stream and filter the data using the [Windows DNS Events via AMA connector](#).

<https://learn.microsoft.com/en-us/azure/sentinel/dns-ama-fields>

ASIM normalized DNS schema

This table describes and translates Windows DNS server fields into the normalized field names as they appear in the [DNS normalization schema](#).

 Expand table

Windows DNS field name	Normalized field name	Type	Description
EventID	EventOriginalType	String	The original event type or ID.
RCODE	EventResult	String	The outcome of the event (success, partial, failure, NA).
RCODE parsed	EventResultDetails	String	The DNS response code as defined by IANA.
InterfaceIP	DvclpAddr	String	The IP address of the event reporting device or interface.
AA	DnsFlagsAuthoritative	Integer	Indicates whether the response from the server was authoritative.
AD	DnsFlagsAuthenticated	Integer	Indicates that the server verified all of the data in the answer and the authority of the response, according to the server policies.
RQNAME	DnsQuery	String	The domain needs to be resolved.
QTYPE	DnsQueryType	Integer	The DNS resource record type as defined by IANA.
Port	SrcPortNumber	Integer	Source port sending the query.
Source	SrclpAddr	IP address	The IP address of the client sending the DNS request. For a recursive DNS request, this value is typically the reporting device's IP, in most cases, <u>127.0.0.1</u> .
ElapsedTime	DnsNetworkDuration	Integer	The time it took to complete the DNS request.
GUID	DnsSessionId	String	The DNS session identifier as reported by the reporting device.

<https://learn.microsoft.com/en-us/azure/sentinel/dns-ama-fields>

130. In maintaining the at least one policy, Microsoft receives, by an adaptive traffic control (ATC) policy management mechanism of an administrative master agent (AMA) in the ATC administrative framework, the at least one policy from at least one source. For instance, Azure Traffic Manager provides adaptive policy-based domain name service (DNS) using Traffic Manager routing methods, which are policies that determine how DNS queries are answered.

Traffic Manager routing methods

✦ Summarize this article for me

Azure Traffic Manager supports six traffic routing methods that determine how to route network traffic to service endpoints, helping you optimize traffic flow and improve application availability. Each Traffic Manager profile applies one routing method to DNS queries: Priority, Weighted, Performance, Geographic, Multivalue, or Subnet. The routing method determines which endpoint to return in the DNS response.

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

131. Traffic Manager routing methods are defined using the Azure Portal.

Manage an Azure Traffic Manager profile

✦ Summarize this article for me

Traffic Manager profiles use traffic-routing methods to control the distribution of traffic to your cloud services or website endpoints. This article explains how to create and manage these profiles.

Create a Traffic Manager profile

You can create a Traffic Manager profile by using the Azure portal. After creating your profile, you can configure endpoints, monitoring, and other settings in the Azure portal. Traffic Manager supports up to 200 endpoints per profile. However, most usage scenarios require only a few of endpoints.

<https://learn.microsoft.com/en-us/azure/traffic-manager/traffic-manager-manage-profiles>

132. In maintaining the at least one policy, Microsoft further broadcasts the received at least one policy to the at least one domain name server in the ATC name server network.

COUNT III: INFRINGEMENT OF THE '964 Patent

133. Plaintiff hereby incorporates by reference each of the allegations in the foregoing

paragraphs as though fully set forth herein, and further alleges as follows.

134. Microsoft directly infringes, either individually or jointly through the direction and control of others, at least claim 19 of the '964 Patent through operation of one or more of the Accused Microsoft Functionalities. For instance, Azure Front Door and/or CDN, and Azure Traffic Manager perform a method operable in a framework wherein resources of a content provider are delivered to clients from one of a plurality of distinct content delivery network domains distinct from a content provider domain.

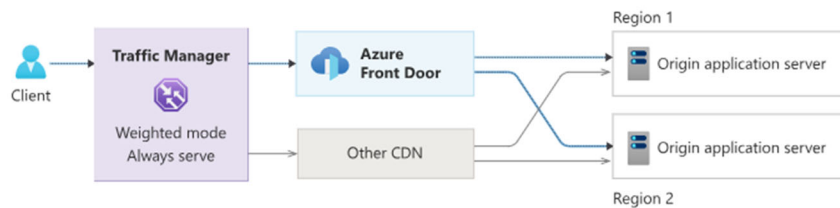
Approach

A third-party CDN can be integrated into your Azure solution to provide isolation from Microsoft's infrastructure. This isolation provides a high degree of resiliency from disaster scenarios. If an outage or disaster occurs, traffic is automatically shifted between Azure Front Door and the alternative CDN. You can use Azure Traffic Manager to detect an outage and redirect traffic to the alternative CDN.

Note

Microsoft offers a CDN interconnect service to route your origin traffic to another CDN with zero data transfer fees. For more information, see [Routing preferences](#).

The following diagram shows how traffic flows between the CDNs:



<https://learn.microsoft.com/en-us/azure/architecture/guide/networking/global-web-applications/mission-critical-content-delivery>

Key features

Increase application availability

Traffic Manager monitors your endpoints and provides automatic failover when an endpoint goes down, delivering high availability for your critical applications.

Improve application performance

Traffic Manager improves your application's responsiveness by directing traffic to the endpoint with the lowest latency for each user.

Service maintenance without downtime

Perform planned maintenance on your applications without downtime. Traffic Manager directs traffic to alternative endpoints while maintenance is in progress.

Combine hybrid applications

Traffic Manager supports external, non-Azure endpoints, enabling hybrid cloud and on-premises deployments including burst-to-cloud, migrate-to-cloud, and failover-to-cloud scenarios.

Distribute traffic for complex deployments

Combine multiple traffic-routing methods with [nested Traffic Manager profiles](#) to create sophisticated and flexible rules that scale to larger, more complex deployments.

<https://learn.microsoft.com/en-us/azure/traffic-manager/traffic-manager-overview>

Mission-critical global content delivery

 Summarize this article for me

Content delivery networks (CDNs) offer a range of capabilities to optimize performance for users, including global layer 7 load balancing and optimized network routing. Caching is also a common way to reduce load on the backend services and provide further resiliency to a range of issues. CDNs, including Azure Front Door, provide caching at the network edge.

CDNs are an essential component in some solution architectures, so it's an industry best practice for mission-critical workloads to use multiple CDNs to achieve a higher level of uptime. If one CDN experiences outage or degraded performance, your traffic is automatically diverted to another CDN.

As mentioned in the previous article in this series, Azure Front Door is designed to provide the utmost resiliency and availability not only for our external customers, but also for multiple properties across Microsoft. While Microsoft offers an [industry-leading service level agreement \(SLA\) for Azure Front Door](#), if you have a mission-critical application that demands even higher SLA, you will need to rely on multiple CDNs. Consider the implications of this approach. Each CDN provides a separate network path to your application servers, and you need to configure and test each CDN separately.

This article describes an approach for using Azure Front Door with another CDN. This approach is suitable for solutions that rely heavily on caching for delivering static content delivery, media, and high-scale eCommerce applications.

<https://learn.microsoft.com/en-us/azure/architecture/guide/networking/global-web-applications/mission-critical-content-delivery>

135. Further, Microsoft obtains content provider policies relating to delivery of resources

where those policies relate to a relative load share for each of the plurality of CDN domains.

Traffic Manager routing methods

 Summarize this article for me

Azure Traffic Manager supports six traffic routing methods that determine how to route network traffic to service endpoints, helping you optimize traffic flow and improve application availability. Each Traffic Manager profile applies one routing method to DNS queries: Priority, Weighted, Performance, Geographic, Multivalue, or Subnet. The routing method determines which endpoint to return in the DNS response.

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

The following traffic routing methods are available in Traffic Manager:

- **Priority:** Select **Priority** routing when you want to use a primary service endpoint for all traffic. You can provide multiple backup endpoints in case the primary endpoint or one of the backup endpoints is unavailable.
- **Weighted:** Select **Weighted** routing when you want to distribute traffic across a set of endpoints based on their weight. Set the weight the same to distribute evenly across all endpoints.
- **Performance:** Select **Performance** routing when you have endpoints in different geographic locations and you want users to use the closest endpoint for the lowest network latency.
- **Geographic:** Select **Geographic** routing to direct users to specific endpoints (Azure, External, or Nested) based on the geographic location of their DNS queries. This routing method helps you comply with scenarios such as data sovereignty mandates, localization of content and user experience, and measuring traffic from different regions.
- **Multivalue:** Select **MultiValue** for Traffic Manager profiles that can only have IPv4/IPv6 addresses as endpoints. When this profile receives a query, it returns all healthy endpoints.
- **Subnet:** Select **Subnet** traffic-routing method to map sets of user IP address ranges to a specific endpoint. When Traffic Manager receives a request, it returns the endpoint mapped to that request's source IP address.

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

136. Microsoft further selects one of the plurality of CDN domains based on the content provider policies, and provides clients with a domain name associated with the selected CDN.

What is Azure Traffic Manager?

✦ Summarize this article for me

Azure Traffic Manager is a DNS-based traffic load balancer that distributes traffic to your public-facing applications across global Azure regions. Traffic Manager provides your public endpoints with high availability and quick responsiveness.

Traffic Manager uses DNS to direct client requests to the appropriate service endpoint based on a traffic-routing method. Traffic Manager also provides health monitoring for every endpoint. The endpoint can be any Internet-facing service hosted inside or outside of Azure. Traffic Manager provides a range of [traffic-routing methods](#) and [endpoint monitoring options](#) to suit different application needs and automatic failover models. Traffic Manager is resilient to failure, including the failure of an entire Azure region.

<https://learn.microsoft.com/en-us/azure/traffic-manager/traffic-manager-overview>

- **Traffic Manager** uses weighted routing mode, has two [endpoints](#), and is configured to [always serve traffic](#).

During normal operations, Traffic Manager sends all requests through Azure Front Door.

If Azure Front Door becomes unavailable, turn off the Azure Front Door endpoint. Traffic Manager then sends all requests through the alternative CDN.

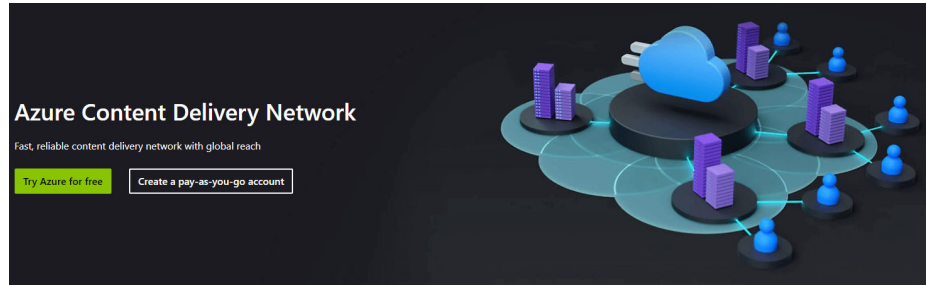
- **Azure Front Door** processes and routes most of your application traffic. Azure Front Door routes traffic to the appropriate origin application server, and it provides the primary path to your application. If Azure Front Door is unavailable, traffic is automatically redirected through the secondary path.
- **An alternative CDN** is configured to send traffic to each origin server.
- **Your origin application servers** need to be ready to accept traffic from both Azure Front Door and another CDN, at any time.

<https://learn.microsoft.com/en-us/azure/architecture/guide/networking/global-web-applications/mission-critical-content-delivery>

COUNT IV: INFRINGEMENT OF THE '899 PATENT

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

138. Microsoft directly infringes, either individually or jointly through the direction and control of others, at least claim 1 of the '899 Patent through operation of one or more of the Accused Microsoft Functionalities. For instance, Azure Front Door and/or CDN provides content delivery systems with a plurality of computers configured to run CD services.



<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>

With Azure CDN from Microsoft, you can monitor resources in the following ways to help you troubleshoot, track, and debug issues.

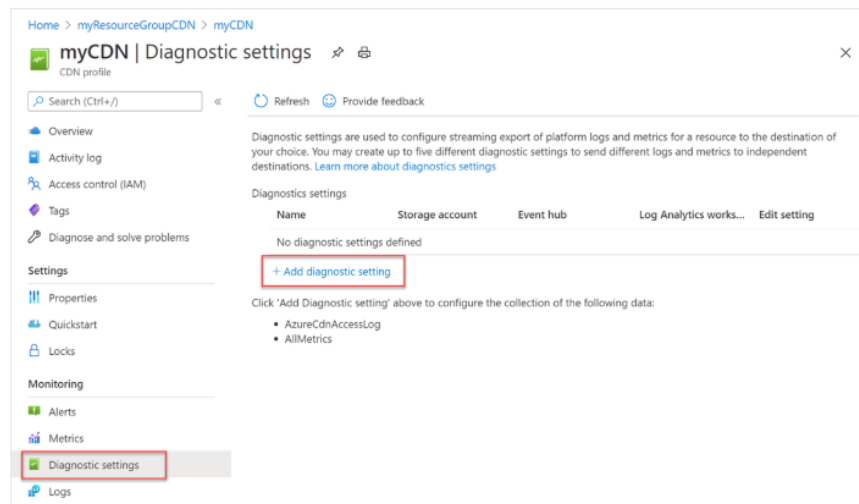
- Raw logs provide rich information about every request that CDN receives. Raw logs differ from activity logs. Activity logs provide visibility into the operations done on Azure resources.
- Metrics, which display four key metrics on CDN, including Byte Hit Ratio, Request Count, Response Size and Total Latency. It also provides different dimensions to break down metrics.
- Alert, which allows customer to set up alert for key metrics
- More metrics, which allow customers to use Azure Log Analytics to enable more metrics of value. We also provide query samples for a few other metrics under Azure Log Analytics.

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

Configuration - Azure portal

To configure Raw logs for your Azure CDN from Microsoft profile:

1. From the Azure portal menu, select **All Resources** >> <your-CDN-profile>.
2. Under **Monitoring**, select **Diagnostics settings**.
3. Select **+ Add diagnostic setting**.



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

Metrics

Azure CDN from Microsoft is integrated with Azure Monitor and publishes four CDN metrics to help track, troubleshoot, and debug issues.

The Metrics are displayed in charts and accessible via PowerShell, CLI, and API. The CDN metrics are free of charge.

Azure CDN from Microsoft measures and sends its metrics in 60-second intervals. The metrics can take up to 3 mins to appear in the portal.

For more information, see [Azure Monitor metrics](#).

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

Metrics supported by Azure CDN from Microsoft

 Expand table

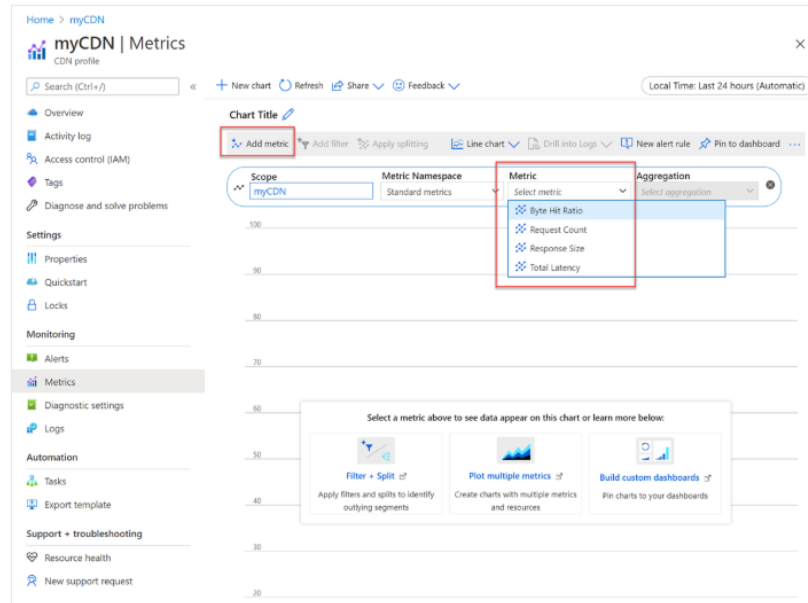
Metrics	Description	Dimensions
Bytes Hit ratio*	The percentage of egress from CDN cache, computed against the total egress.	Endpoint
RequestCount	The number of client requests served by CDN.	Endpoint Client country. Client region. HTTP status. HTTP status group.
ResponseSize	The number of bytes sent as responses from CDN edge to clients.	Endpoint Client country. Client region. HTTP status. HTTP status group.
TotalLatency	The total time from the client request received by CDN until the last response byte send from CDN to client.	Endpoint Client country. Client region. HTTP status. HTTP status group.

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

139. These CD services produce event output relating to their respective operations, and that output is provided to CD services. Microsoft's Azure Front Door and/or CDN also obtains event input from CD services, and produces, as respective state output, state information based on the event input from the CD services.

Metrics configuration

1. From the Azure portal menu, select **All Resources** >> <your-CDN-profile>.
2. Under **Monitoring**, select **Metrics**:
3. Select **Add metric**, select the metric to add:



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

Metrics

Azure CDN from Microsoft is integrated with Azure Monitor and publishes four CDN metrics to help track, troubleshoot, and debug issues.

The Metrics are displayed in charts and accessible via PowerShell, CLI, and API. The CDN metrics are free of charge.

Azure CDN from Microsoft measures and sends its metrics in 60-second intervals. The metrics can take up to 3 mins to appear in the portal.

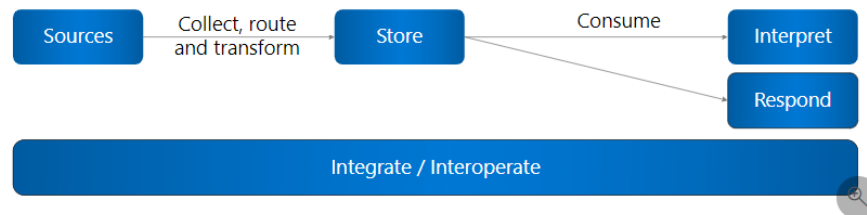
For more information, see [Azure Monitor metrics](#).

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

140. Further, Microsoft's Azure Front Door and/or CDN maintains service configuration information specifying associations between CD service instances, obtains state information from said CD service instances, and modifies said service configuration information to produce modified service configuration information based on the state information.

Azure Monitor is a comprehensive monitoring solution for collecting, analyzing, and responding to monitoring data from your cloud and on-premises environments. You can use Azure Monitor to maximize the availability and performance of your applications and services. It helps you understand how your applications are performing and allows you to manually and programmatically respond to system events.

Azure Monitor collects and aggregates the data from every layer and component of your system across multiple Azure and non-Azure subscriptions and tenants. It stores it in a common data platform for consumption by a common set of tools that can correlate, analyze, visualize, and/or respond to the data. You can also integrate other Microsoft and non-Microsoft tools.



The diagram above shows an abstracted view of the monitoring process. A more detailed breakdown of the Azure Monitor architecture is shown in the [High level architecture](#) section below.

<https://learn.microsoft.com/en-us/azure/azure-monitor/overview>

High level architecture

Azure Monitor can monitor these types of resources in Azure, other clouds, or on-premises:

- Applications
- Virtual machines
- Guest operating systems
- Containers including Prometheus metrics
- Databases
- Security events in combination with Azure Sentinel
- Networking events and health in combination with Network Watcher
- Custom sources that use the APIs to get data into Azure Monitor

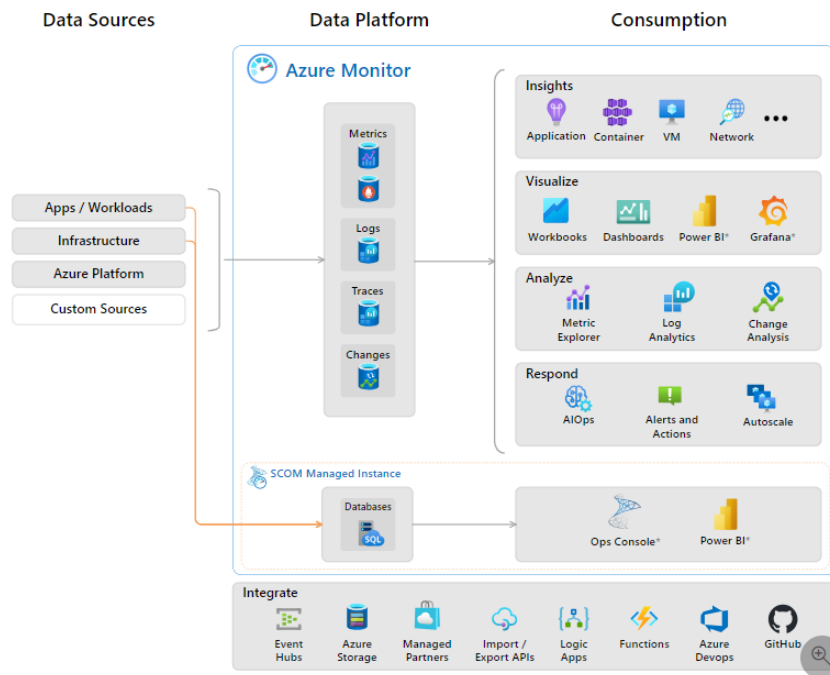
You can also export monitoring data from Azure Monitor into other systems so you can:

- Integrate with other third-party and open-source monitoring and visualization tools
- Integrate with ticketing and other ITSM systems

If you're a System Center Operations Manager (SCOM) user, Azure Monitor now includes Azure Monitor [SCOM Managed Instance \(SCOM MI\)](#). Operations Manager MI is a cloud-hosted version of Operations Manager and allows you to move your on-premises Operations Manager installation to Azure.

<https://learn.microsoft.com/en-us/azure/azure-monitor/overview>

The following diagram shows a high-level architecture view of Azure Monitor.

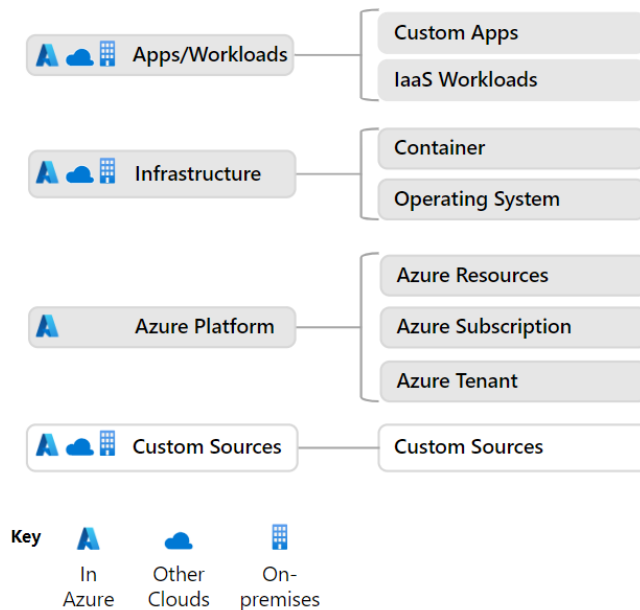


<https://learn.microsoft.com/en-us/azure/azure-monitor/overview>

Data sources

Azure Monitor can collect data from multiple sources.

The diagram below shows an expanded version of the data source types that Azure Monitor can gather monitoring data from.

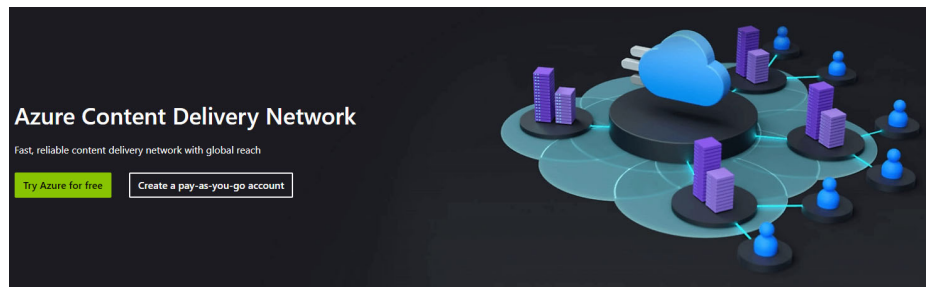


<https://learn.microsoft.com/en-us/azure/azure-monitor/overview>

COUNT V: INFRINGEMENT OF THE '876 PATENT

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

142. Microsoft directly infringes, either individually or jointly through the direction and control of others, at least claim 1 of the '876 Patent through operation of one or more of the Accused Microsoft Functionalities. For instance, Microsoft Azure Front Door and/or CDN, Azure Monitor, Azure Peering Service comprise a CDN with multiple CD services with a collector system.



<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.

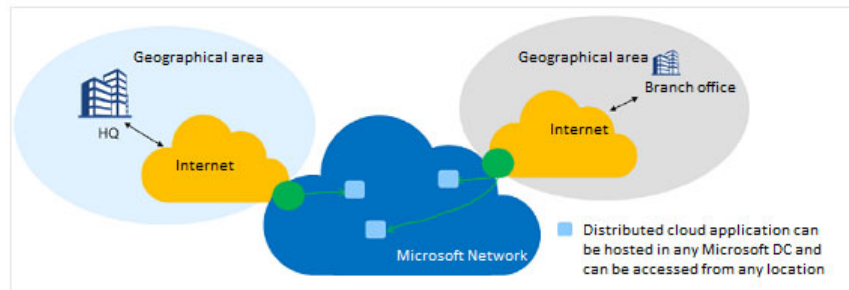


Integrate seamlessly with your Azure services to activate within minutes.



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

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



Customers can also opt for Peering Service telemetry such as user latency measures to the Microsoft network and BGP route monitoring by registering the Peering Service connection in the Azure portal.

To use Peering Service, customers aren't required to register with Microsoft. The only requirement is to contact a [Peering Service partner](#) to get the service. To opt in for Peering Service telemetry, customers must register for it in the Azure portal.

For instructions on how to register a Peering Service, see [Create, change, or delete a Peering Service connection using the Azure portal](#).

<https://learn.microsoft.com/en-us/azure/peering-service/about>

What is Peering Service?

Peering Service is:

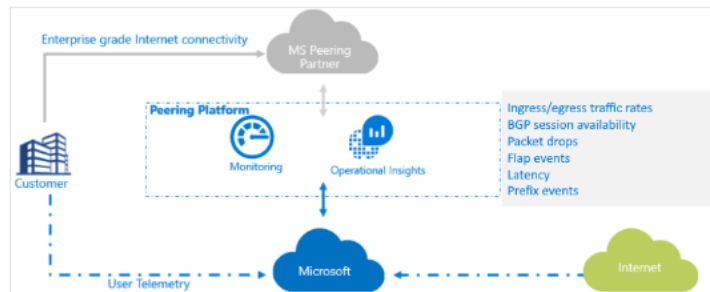
- An IP service that uses the public internet.
- A collaboration platform with service providers and a value-added service that's intended to offer optimal and reliable routing via service provider partners to the Microsoft cloud over the public network.

<https://learn.microsoft.com/en-us/azure/peering-service/about>

Monitoring platform

Service monitoring is offered to analyze user traffic and routing. The following metrics are available in the Azure portal to track the performance and availability of your Peering Service connection:

- Ingress and egress traffic rates
- BGP session availability
- Packet drops
- Flap events
- Latency
- Prefix events



<https://learn.microsoft.com/en-us/azure/peering-service/about>

143. Azure Front Door and/or CDN, Azure Monitor, Azure Peering Service further receive multiple event streams of event data, said multiple event streams comprising event data from a plurality of CD services in the CDN, and said plurality of CDN services including a first CD services, each event of said event stream comprising a timestamp for said event and information relating to said event.

View metrics for an internet peering

Article • 11/22/2024 • 1 contributor [Feedback](#)

In this article

- [View received routes](#)
- [View peering metrics](#)
- [View registered prefix latency](#)
- [View customer prefix latency](#)
- [Show 2 more](#)

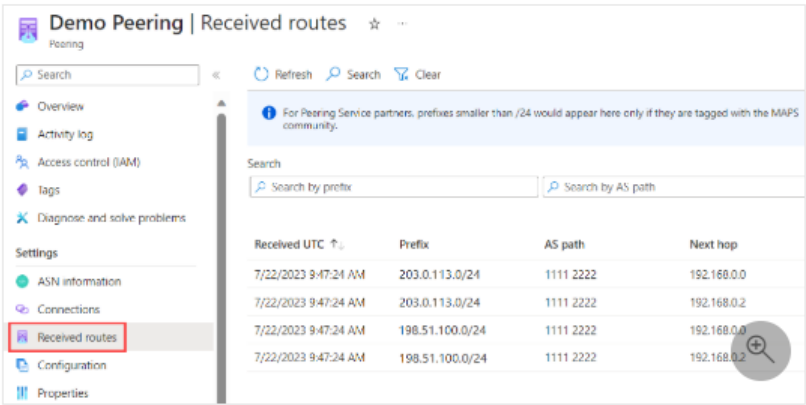
In this article, as an internet peering partner (direct or exchange) in Azure Peering Service, you learn how to use the Azure portal to view metrics associated with a direct peering or an exchange peering.

<https://learn.microsoft.com/en-us/azure/internet-peering/walkthrough-monitoring-telemetry>

View received routes

You can view routes that are announced to Microsoft by using the Azure portal.

1. In the Azure portal, go to the peering.
2. On the service menu under **Settings**, select **Received Routes**.



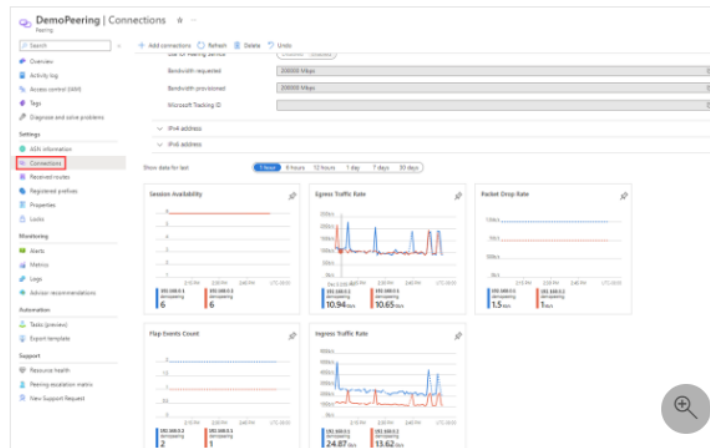
<https://learn.microsoft.com/en-us/azure/internet-peering/walkthrough-monitoring-telemetry>

144. Microsoft further produces state data relating to and based information represented in said event data of the multiple event streams, and asynchronously responds to queries relating to said state data.

View peering metrics

You can view the following metrics for a peering on the **Connections** pane of a peering:

- Session availability
- Ingress traffic rate
- Egress traffic rate
- Flap events count
- Packet drop rate

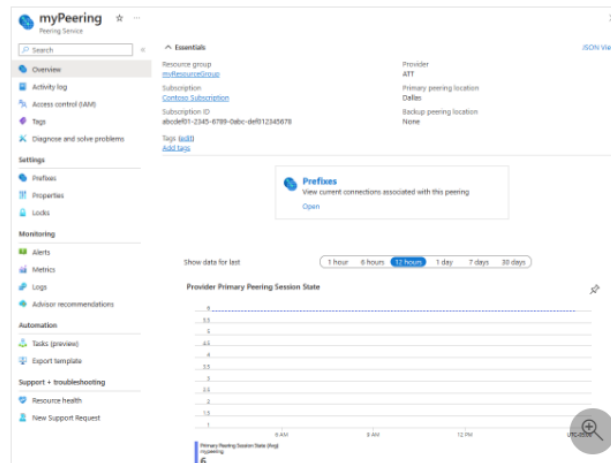


<https://learn.microsoft.com/en-us/azure/internet-peering/walkthrough-monitoring-telemetry>

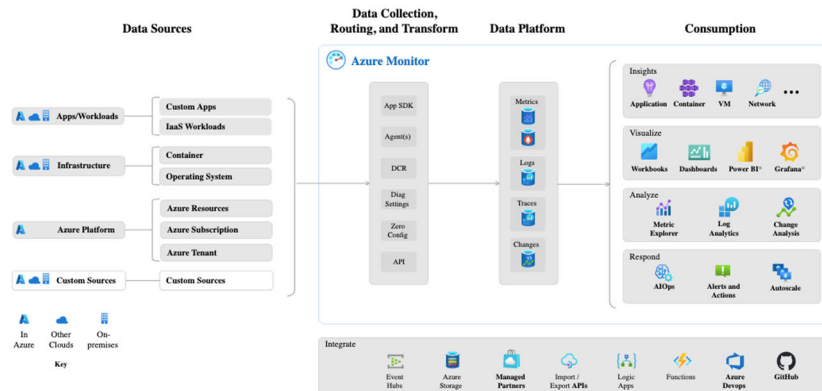
View Peering Service metrics

All Peering Service resources display the session availability metrics for their Peering Service on the overview pane of a Peering Service resource:

- **Provider primary peering session availability:** Indicates the state of the Border Gateway Protocol (BGP) session between the Peering Service provider and Microsoft at the primary peering location.
- **Provider backup peering session availability:** Indicates the state of the BGP session between the Peering Service provider and Microsoft at the backup peering location if one is selected for the Peering Service resource.



<https://learn.microsoft.com/en-us/azure/internet-peering/walkthrough-monitoring-telemetry>



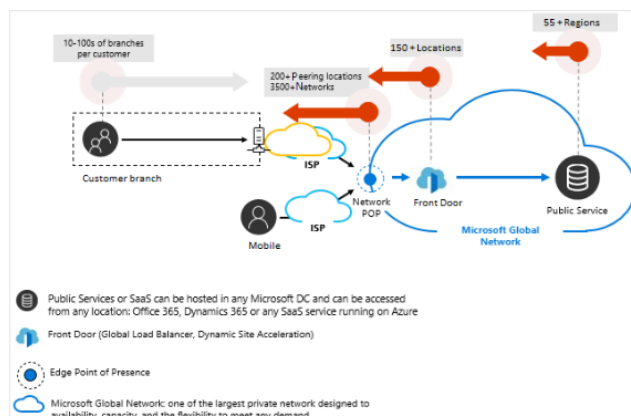
<https://learn.microsoft.com/en-us/azure/azure-monitor/fundamentals/data-platform>

145. Microsoft further, in response to queries from CD services relating to said state data, provides said state data to said first CD service, wherein said state data are used to inform a peering policy of a set of peer caches.

Background

Microsoft 365, Dynamics 365, and any other Microsoft SaaS services are hosted in multiple Microsoft datacenters and can be accessed from any geographic location. The Microsoft global network has Microsoft Edge point-of-presence (PoP) locations around the world where it can connect to an end user via their service providers.

Microsoft and partner service providers ensure that the traffic for the prefixes registered with a Peering Service connection enters and exits the nearest Microsoft Edge PoP locations on the Microsoft global network. Microsoft ensures that the networking traffic egressing from the prefixes registered with Peering Service connections takes the nearest Microsoft Edge PoP locations on the Microsoft global network.



<https://learn.microsoft.com/en-us/azure/peering-service/about>

Why use Peering Service?

Enterprises looking for internet-first access to the cloud or considering SD-WAN architecture or with high usage of Microsoft SaaS services need robust and high-performing internet connectivity. Customers can make that transition happen by using Peering Service. Microsoft and service providers partnered to deliver reliable and performance-centric public connectivity to the Microsoft cloud. Some of the key customer features are listed here:

- Best public routing over the internet to Microsoft Azure Cloud Services for optimal performance and reliability.
- Ability to select the preferred service provider to connect to the Microsoft cloud.
- Traffic insights such as latency reporting and prefix monitoring.
- Optimum network hops (AS hops) from the Microsoft cloud.
- Route analytics and statistics: Events for BGP route anomalies and suboptimal routing.

<https://learn.microsoft.com/en-us/azure/peering-service/about>

Routing architecture overview



<https://learn.microsoft.com/en-us/azure/frontdoor/front-door-routing-architecture>

Delivery of 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, Azure Resource Manager templates, the Azure CLI, and PowerShell.
- Define your own *custom domain* with flexible domain validation.
- Load balance and route traffic across *origins*. Use intelligent *health probe* monitoring across apps or content hosted in Azure or anywhere.
- Integrate with other Azure services, such as Azure DNS, the Web Apps feature of Azure App Service, Azure 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 Azure Front Door and security patterns.
- *Monitor your Azure Front Door traffic in real time*, and configure alerts that integrate with Azure Monitor.
- *Log each Azure Front Door request* and failed health probes.

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

COUNT VI: INFRINGEMENT OF THE '066 PATENT

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

147. Microsoft directly infringes, either individually or jointly through the direction and control of others, at least claim 14 of the '066 Patent through operation of one or more of the Accused Microsoft Functionalities. For instance, Azure Web Application Firewall (“WAF”) functionality is implemented in software running on Microsoft’s global edge network and evaluates incoming requests against configured rules. For instance, Web Application Firewall receives content requests including first sets of attribute values and uses at least one of the attribute values from the first set of attribute values to determine a second set of attribute values.

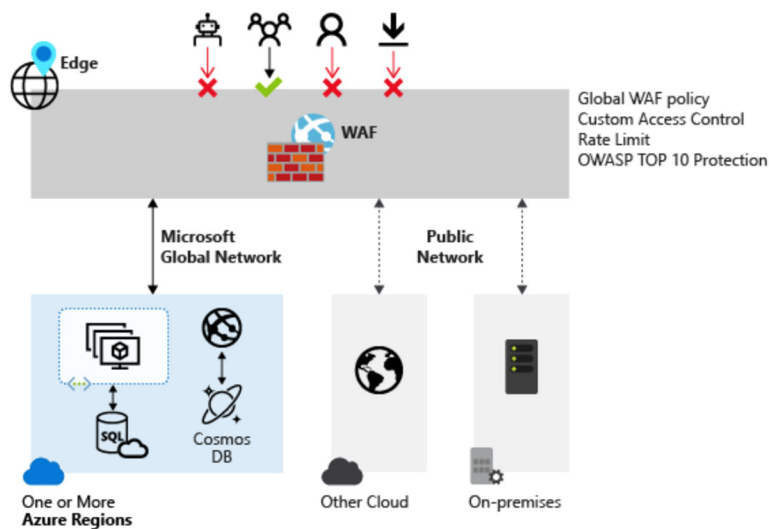
What is Azure Web Application Firewall?

✦ Summarize this article for me

Applies to: ✓ Application Gateway ✓ Front Door Standard/Premium ✓ Front Door (classic) ✓ CDN Standard from Microsoft (classic)

Azure Web Application Firewall provides centralized protection of your web applications from common exploits and vulnerabilities. Web applications increasingly encounter malicious attacks that exploit commonly known vulnerabilities. SQL injection and cross-site scripting are among the most common attacks.

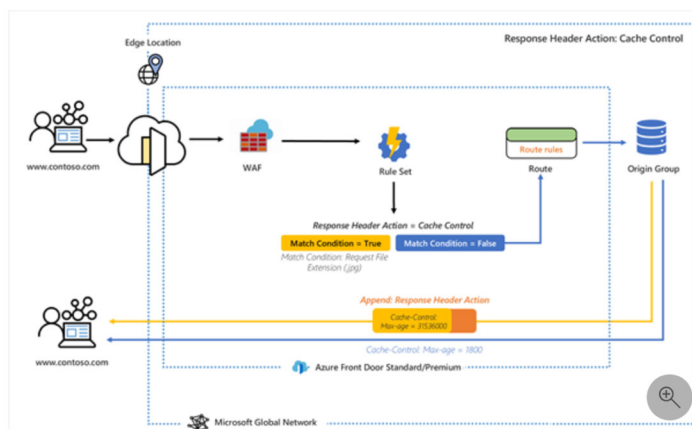
<https://learn.microsoft.com/en-us/azure/web-application-firewall/overview>



<https://learn.microsoft.com/en-us/azure/web-application-firewall/overview>

Example

In the following diagram, WAF policies get processed first. Then the rule set configuration appends a response header. The header changes the max-age of the cache control if the match condition is true.



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

- **Match condition**

Defines a match variable, an operator, and a match value. Each rule can contain multiple match conditions. A match condition might be based on **geo**-location, client IP addresses (CIDR), size, AS Number, Client Fingerprint, Service Tag, or string match. String match can be against a list of match variables.

<https://learn.microsoft.com/en-us/azure/web-application-firewall/afds/waf-front-door-custom-rules>

Remote address

The **remote address** match condition identifies requests based on the requester's location or IP address. You can specify multiple values to match, which can be combined using OR logic.

- Use CIDR notation when specifying IP address blocks. The syntax for an IP address block is the base IP address followed by a forward slash and the prefix size. For example:
 - **IPv4 example:** `5.5.5.64/26` matches any requests that arrive from addresses 5.5.5.64 through 5.5.5.127.
 - **IPv6 example:** `1:2:3::/48` matches any requests that arrive from addresses 1:2:3:0:0:0:0:0 through 1:2:3:ffff:ffff:ffff:ffff:ffff.
- When you specify multiple IP addresses and IP address blocks, 'OR' logic is applied.
 - **IPv4 example:** if you add two IP addresses `1.2.3.4` and `10.20.30.40`, the condition is matched for any requests that arrive from either address 1.2.3.4 or 10.20.30.40.
 - **IPv6 example:** if you add two IP addresses `1:2:3:4:5:6:7:8` and `10:20:30:40:50:60:70:80`, the condition is matched for any requests that arrive from either address 1:2:3:4:5:6:7:8 or 10:20:30:40:50:60:70:80.
- The remote address represents the original client IP that is either from the network connection or typically the X-Forwarded-For request header if the user is behind a proxy. Use the **socket address** match condition (available in Standard/Premium), if you need to match based on the TCP request's IP address.

<https://learn.microsoft.com/en-us/azure/frontdoor/rules-match-conditions>

<code>hostname</code>	The host name in the request from the client. To access this server variable in a match condition, use Host name .
<code>geo_country</code>	Indicates the requester's country/region of origin through its country/region code. To access this server variable in a match condition, use Remote address and configure the <i>Operator</i> to <i>Geo Match</i> or <i>Geo Not Match</i> .
<code>http_method</code>	The method used to make the URL request, such as GET or POST . To access this server variable in a match condition, use Request method .

<https://learn.microsoft.com/en-us/azure/frontdoor/rule-set-server-variables>

148. Web Application Firewall further traverses a hierarchy of decision nodes, where each decision node implements business logic based on one of the attribute values from the first set of

attribute values or the second set of attribute values; and yields a decision from a last node in the hierarchy, wherein the decision dictates how to respond to the content request.

Priority, action types, and match conditions

You can control access with a custom WAF rule that defines a priority number, a rule type, an array of match conditions, and an action.

- **Priority**

A unique integer that describes the order of evaluation of WAF rules. Rules with lower-priority values are evaluated before rules with higher values. The rule evaluation stops on any rule action except for *Log*. Priority numbers must be unique among all custom rules.

- **Action**

Defines how to route a request if a WAF rule is matched. You can choose one of the following actions to apply when a request matches a custom rule.

- **Allow:** The WAF allows the request to process, logs an entry in WAF logs, and exits.
- **Block:** Request is blocked. The WAF sends a response to a client without forwarding the request further. The WAF logs an entry in WAF logs and exits.
- **Log:** The WAF logs an entry in WAF logs and continues to evaluate the next rule in the priority order.
- **Redirect:** The WAF redirects the request to a specified URI, logs an entry in WAF logs, and exits.

- **Match condition**

Defines a match variable, an operator, and a match value. Each rule can contain multiple match conditions. A match condition might be based on geo-location, client IP addresses (CIDR), size, AS Number, Client Fingerprint, Service Tag, or string match. String match can be against a list of match variables.

<https://learn.microsoft.com/en-us/azure/web-application-firewall/afds/waf-front-door-custom-rules>

Use custom rules

After you identify what's causing a WAF rule match, you can use custom rules to adjust how the WAF responds to the event. Custom rules are processed before managed rules. They can contain more than one condition, and their actions can be [Allow](#), [Deny](#), [Log](#), or [Redirect](#).

Warning

When a request matches a custom rule, the WAF engine stops processing the request. Managed rules won't be processed for this request and neither will other custom rules with a lower priority.

The following example shows a custom rule with two conditions. The first condition looks for the `comment` value in the request body. The second condition looks for the `/api/Feedbacks/` value in the request URI.

<https://learn.microsoft.com/en-us/azure/web-application-firewall/afds/waf-front-door-tuning>

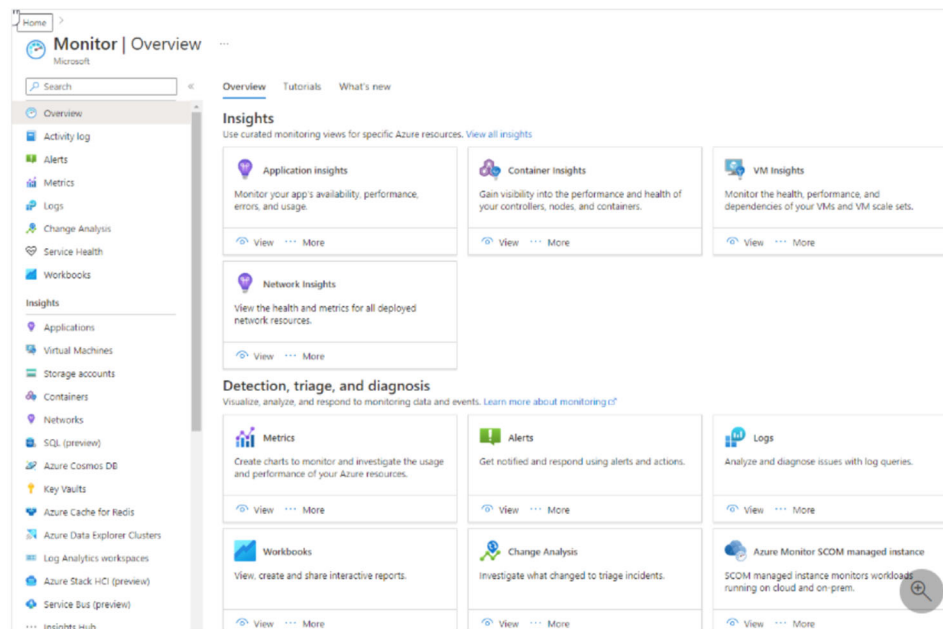
COUNT VII: INFRINGEMENT OF THE '738 PATENT

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

150. Microsoft directly infringes, either individually or jointly through the direction and control of others, at least claim 1 of the '738 Patent through operation of one or more of the Accused Microsoft Functionalities. For instance, Azure Traffic Manager, Azure Monitor, and Azure Service Health receive from subscribers, via a web-based graphical user interface, one or more policies for distribution of service requests to one or more servers in a subscriber server network.

The Azure portal

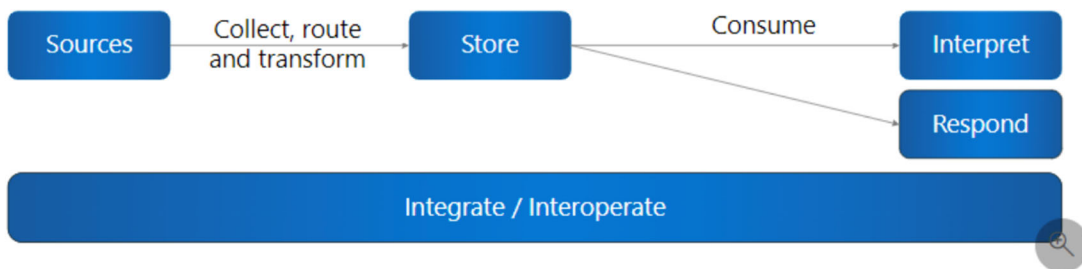
The Azure portal is a web-based, unified console that provides an alternative to command-line tools. With the Azure portal, you can manage your Azure subscription using a graphical user interface. You can build, manage, and monitor everything from simple web apps to complex cloud deployments in the portal. The *Monitor* section of the Azure portal provides a visual interface that gives you access to the data collected for Azure resources and an easy way to access the tools, insights, and visualizations in Azure Monitor.



<https://learn.microsoft.com/en-us/azure/azure-monitor/overview>

Azure Monitor is a comprehensive monitoring solution for collecting, analyzing, and responding to monitoring data from your cloud and on-premises environments. You can use Azure Monitor to maximize the availability and performance of your applications and services. It helps you understand how your applications are performing and allows you to manually and programmatically respond to system events.

Azure Monitor collects and aggregates the data from every layer and component of your system across multiple Azure and non-Azure subscriptions and tenants. It stores it in a common data platform for consumption by a common set of tools that can correlate, analyze, visualize, and/or respond to the data. You can also integrate other Microsoft and non-Microsoft tools.



The diagram above shows an abstracted view of the monitoring process. A more detailed breakdown of the Azure Monitor architecture is shown in the [High level architecture](#) section below.

<https://learn.microsoft.com/en-us/azure/azure-monitor/overview>

High level architecture

Azure Monitor can monitor these types of resources in Azure, other clouds, or on-premises:

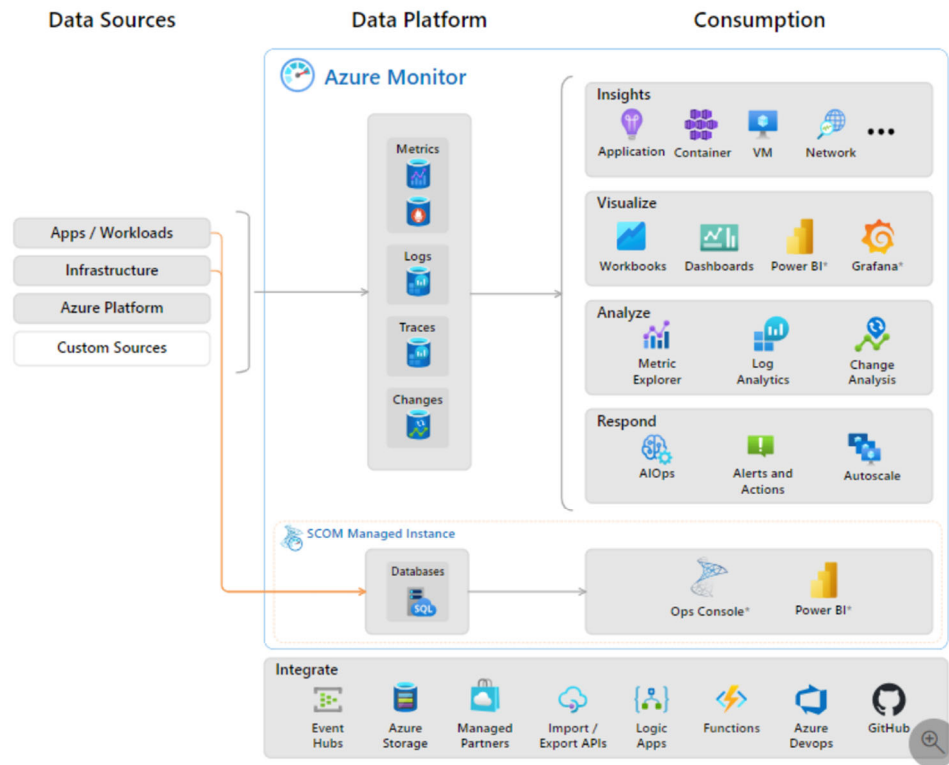
- Applications
- Virtual machines
- Guest operating systems
- Containers including Prometheus metrics
- Databases
- Security events in combination with Azure Sentinel
- Networking events and health in combination with Network Watcher
- Custom sources that use the APIs to get data into Azure Monitor

You can also export monitoring data from Azure Monitor into other systems so you can:

- Integrate with other third-party and open-source monitoring and visualization tools
- Integrate with ticketing and other ITSM systems

If you're a System Center Operations Manager (SCOM) user, Azure Monitor now includes Azure Monitor [SCOM Managed Instance \(SCOM MI\)](#). Operations Manager MI is a cloud-hosted version of Operations Manager and allows you to move your on-premises Operations Manager installation to Azure.

<https://learn.microsoft.com/en-us/azure/azure-monitor/overview>



<https://learn.microsoft.com/en-us/azure/azure-monitor/overview>

151. Azure Traffic Manager, Azure Monitor, and Azure Service Health further monitor operational statuses of some of a plurality of name servers in a name server network that provide hostname resolution service for the one more server in the subscriber server network. Azure further, based on said monitoring, when an error or anomalous condition is detected at a particular name server of the name server network, creates policies to indicate the status of the particular name server, wherein the policy is taken into account when resolving hostnames using the name server network.

Data Type	Description and subtypes
App/Workloads	App- Application performance, health, and activity data. Workloads - IaaS workloads such as SQL server, Oracle or SAP running on a hosted Virtual Machine.
Infrastructure	Container - Data about containers, such as Azure Kubernetes Service, Prometheus, and the applications running inside containers. Operating system - Data about the guest operating system on which your application is running.
Azure Platform	Azure resource - Data about the operation of an Azure resource from inside the resource, including changes. Resource Logs are one example. Azure subscription - The operation and management of an Azure subscription, and data about the health and operation of Azure itself. The activity log is one example. Azure tenant - Data about the operation of tenant-level Azure services, such as Microsoft Entra ID.
Custom Sources	Data that gets into the system using the - Azure Monitor REST API - Data Collection API

<https://learn.microsoft.com/en-us/azure/azure-monitor/overview>

Collection method	Description
Application instrumentation	Application Insights is enabled through either Auto-Instrumentation (agent) or by adding the Application Insights SDK to your application code. In addition, Application Insights is in process of implementing Open Telemetry . For more information, reference How do I instrument an application? .
Agents	Agents can collect monitoring data from the guest operating system of Azure and hybrid virtual machines.
Data collection rules	Use data collection rules to specify what data should be collected, how to transform it, and where to send it.
Zero Config	Data is automatically sent to a destination without user configuration. Platform metrics are the most common example.
Diagnostic settings	Use diagnostic settings to determine where to send resource log and activity log data on the data platform.
Azure Monitor REST API	The Logs Ingestion API in Azure Monitor lets you send data to a Log Analytics workspace in Azure Monitor Logs. You can also send metrics into the Azure Monitor Metrics store using the custom metrics API.

<https://learn.microsoft.com/en-us/azure/azure-monitor/overview>

Monitor Azure DNS

Article • 01/30/2025 • 2 contributors

[Feedback](#)

In this article

[Collect data with Azure Monitor](#)

[Use Azure Monitor tools to analyze the data](#)

[Export Azure Monitor data](#)

[Use Kusto queries to analyze log data](#)

[Show 2 more](#)

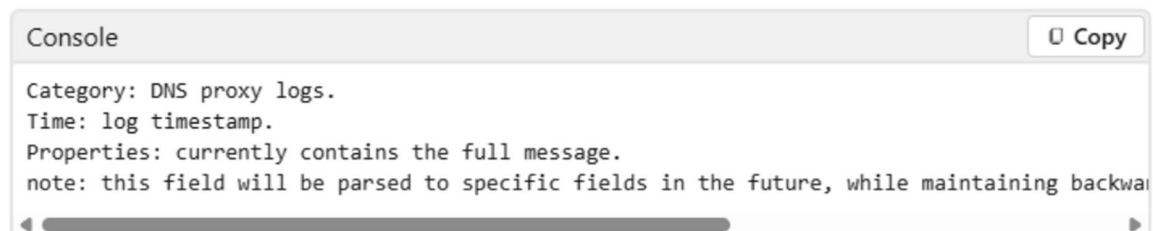
Azure Monitor collects and aggregates metrics and logs from your system to monitor availability, performance, and resilience, and notify you of issues affecting your system. You can use the Azure portal, PowerShell, Azure CLI, REST API, or client libraries to set up and view monitoring data.

Different metrics and logs are available for different resource types. This article describes the types of monitoring data you can collect for this service and ways to analyze that data.

<https://learn.microsoft.com/en-us/azure/dns/monitor-dns>

DNS proxy log

The DNS proxy log is saved to a storage account, streamed to Event hubs, and/or sent to Azure Monitor logs only if you enable it for each Azure Firewall. This log tracks DNS messages to a DNS server configured using DNS proxy. The data is logged in JSON format, as shown in the following examples:



```
Console Copy
Category: DNS proxy logs.
Time: log timestamp.
Properties: currently contains the full message.
note: this field will be parsed to specific fields in the future, while maintaining backward compatibility
```

<https://learn.microsoft.com/en-us/azure/firewall/monitor-firewall>

Service Health notifications



Summarize this article for me

In this article

[View Service Health notifications](#)

[Service Health notification types](#)

[Related content](#)

Azure Service Health notifications are system-generated alerts that inform you about Azure service problems or events that affect your resources. The subscription's [Azure activity log](#) records these notifications as part of logging many events in Azure. The Azure portal then displays them under [Azure Service Health](#).

When Azure needs to communicate something about service health, such as an outage, upcoming maintenance, or an account-specific alert, it creates a Service Health event in your activity log.

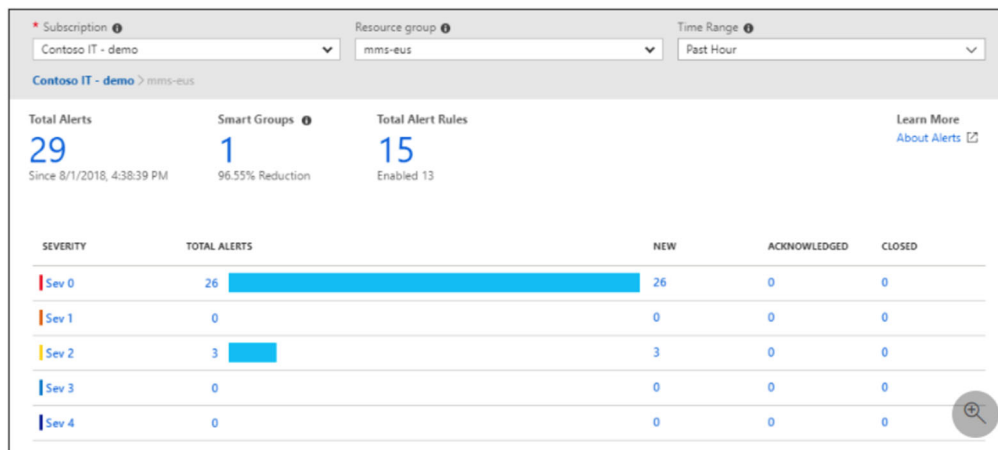
<https://learn.microsoft.com/en-us/azure/service-health/service-health-notifications-properties>

152. Azure Traffic manager further sends alerts regarding the error or anomalous condition detected at a particular name server to the subscriber via the web-based graphical user interface.

Azure Monitor Alerts notify you of critical conditions and can take corrective action. Alert rules can be based on metric or log data.

- Metric alert rules provide near-real-time alerts based on collected metrics.
- Log search alert rules based on logs allow for complex logic across data from multiple sources.

Alert rules use **action groups**, which can perform actions such as sending email or SMS notifications. Action groups can send notifications using webhooks to trigger external processes or to integrate with your IT service management tools. Action groups, actions, and sets of recipients can be shared across multiple rules.

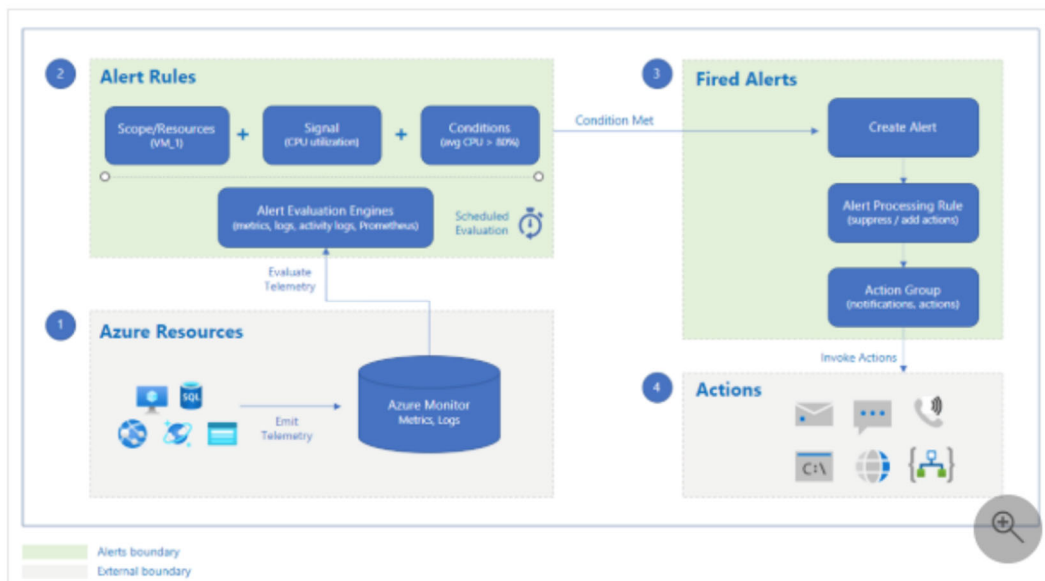


<https://learn.microsoft.com/en-us/azure/azure-monitor/overview>

Alerts help you detect and address issues before users notice them by proactively notifying you when Azure Monitor data indicates there might be a problem with your infrastructure or application.

You can alert on any metric or log data source in the Azure Monitor data platform.

This diagram shows you how alerts work.



<https://learn.microsoft.com/en-us/azure/azure-monitor/alerts/alerts-overview>

Post Incident Review (PIR) – Azure Storage – Unreachable Blob and Data Lake storage accounts in multiple regions

Tracking ID: PSM0-BQ8



Watch our 'Azure Incident Retrospective' video about this incident: <https://aka.ms/AIR/PSM0-BQ8>

What happened?

Between 00:50 UTC and 12:30 UTC on 13 November 2024, a subset of Azure Blob Storage and Azure Data Lake Storage accounts experienced connectivity errors. The issue manifested as loss of access to Blob and Data Lake storage endpoints of the affected storage accounts, and subsequent unavailability of downstream services that depended on these storage accounts. Since many of the impacted storage accounts were used by other Azure services and major software vendor solutions, the customer impact was widespread. Although unavailable to access, the data stored in these storage accounts was not impacted during this incident. Impacted downstream services included:

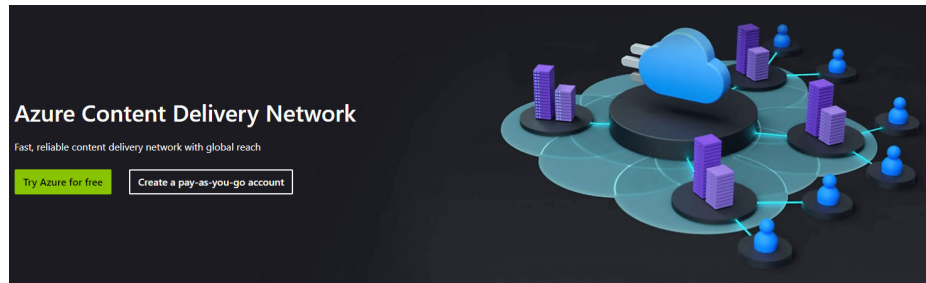
- **Azure Storage:** Impacted customers may have experienced name (DNS) resolution failures when interacting with impacted storage accounts in Australia East, Australia Southeast, Brazil South, Brazil Southeast, Canada Central, Canada East, Central India, Central US, East Asia, East US, East US 2, East US 2 EUAP, France Central, Germany West Central, Japan East, Japan West, Korea Central, North Central US, North Europe, Norway East, South Africa North, South Central US, South India, Southeast Asia, Sweden Central, Switzerland North, UAE North, UK South, UK West, West Central US, West Europe, West US, West US 2, West US 3.
- **Azure Container Registry:** Impacted customers using the East US region may have experienced intermittent 5xx errors while trying to pull images from the registry.
- **Azure Databricks:** Impacted customers may have experienced failures with launching clusters and serverless compute resources in Australia East, Canada Central, Canada East, Central US, East US, East US 2, Japan East, South Central US, UAE North, West US, and/or West US 2.
- **Azure Log Analytics:** Impacted customers using the West Europe, Southeast Asia, and/or Korea Central regions may have experienced delays and/or stale data when viewing Microsoft Graph activity logs.

<https://azure.status.microsoft.com/status/history/?trackingId=PSM0-BQ8>

COUNT VIII: INFRINGEMENT OF THE '517 PATENT

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

154. Microsoft directly infringes, either individually or jointly through the direction and control of others, at least claim 1 of the '517 Patent through operation of one or more of the Accused Microsoft Functionalities. For instance, Azure Front Door and Azure CDN, and Azure Traffic Manager provide an adaptive traffic control network.



<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>

What is Traffic Manager?

Article • 08/15/2023 • 12 contributors

[Feedback](#)

In this article

- Increase application availability
- Improve application performance
- Service maintenance without downtime
- Combine hybrid applications

[Show 3 more](#)

Azure Traffic Manager is a DNS-based traffic load balancer. This service allows you to distribute traffic to your public facing applications across the global Azure regions. Traffic Manager also provides your public endpoints with high availability and quick responsiveness.

Traffic Manager uses DNS to direct client requests to the appropriate service endpoint based on a traffic-routing method. Traffic manager also provides health monitoring for every endpoint. The endpoint can be any Internet-facing service hosted inside or outside of Azure. Traffic Manager provides a range of [traffic-routing methods](#) and [endpoint monitoring options](#) to suit different application needs and automatic failover models. Traffic Manager is [resilient](#) to failure, including the failure of an entire Azure region.

<https://learn.microsoft.com/en-us/azure/traffic-manager/traffic-manager-overview>

Traffic Manager routing methods

Article • 08/08/2024 • 15 contributors

[Feedback](#)

In this article

[Priority traffic-routing method](#)
[Weighted traffic-routing method](#)
[Performance traffic-routing method](#)
[Geographic traffic-routing method](#)
[Show 3 more](#)

Azure Traffic Manager supports six traffic-routing methods to determine how to route network traffic to the various service endpoints. For any profile, Traffic Manager applies the traffic-routing method associated to it to each DNS query it receives. The traffic-routing method determines which endpoint is returned in the DNS response.

The following traffic routing methods are available in Traffic Manager:

- **Priority:** Select **Priority** routing when you want to have a primary service endpoint for all traffic. You can provide multiple backup endpoints in case the primary or one of the backup endpoints is unavailable.
- **Weighted:** Select **Weighted** routing when you want to distribute traffic across a set of endpoints based on their weight. Set the weight the same to distribute evenly across all endpoints.
- **Performance:** Select **Performance** routing when you have endpoints in different geographic locations and you want end users to use the "closest" endpoint for the lowest network latency.
- **Geographic:** Select **Geographic** routing to direct users to specific endpoints (Azure, External, or Nested) based on where their DNS queries originate from geographically. With this routing method, it enables you to be in compliance with scenarios such as data sovereignty mandates, localization of content & user experience and measuring traffic from different regions.
- **MultiValue:** Select **MultiValue** for Traffic Manager profiles that can only have IPv4/IPv6 addresses as endpoints. When a query is received for this profile, all healthy endpoints are returned.
- **Subnet:** Select **Subnet** traffic-routing method to map sets of end-user IP address ranges to a specific endpoint. When a request is received, the endpoint returned will be the one mapped for that request's source IP address.

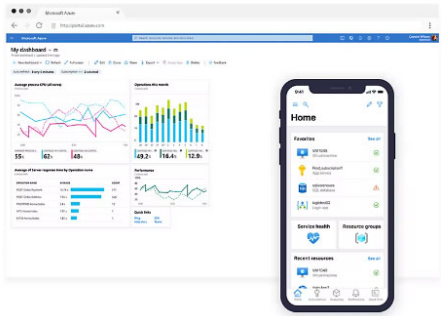
<https://learn.microsoft.com/en-us/azure/traffic-manager/traffic-manager-routing-methods>

155. Azure Front Door/CDN and Traffic Manager performs a method in which an adaptive traffic control name server network implements policy-based traffic direction, where Azure Front Door/CDN and Traffic Manager provides a graphical user interface (GUI) and uses said GUI to obtain at least one policy for direction of network traffic, wherein the GUI supports the setting of said at least one policy using a decision tree representing rules.

View one portal, manage all
your apps

View and manage all of your applications in one unified hub—including web apps, databases, virtual machines, virtual networks, storage, and Visual Studio team projects. Enjoy the flexibility of using the Azure portal's graphical experience or the integrated command-line experience provided by [Cloud Shell](#). [Get an overview](#), and [see which Azure services can be managed with the Azure portal](#).

<https://azure.microsoft.com/en-us/get-started/azure-portal>



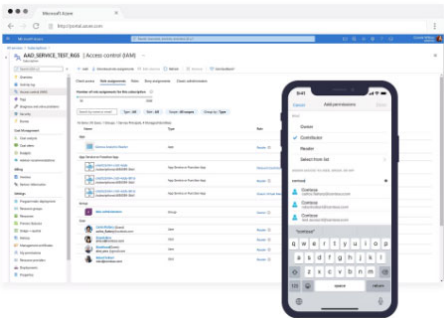
Personalize your experience

Imagine a single easy-to-use console built just for you, your team, and your projects. This unified hub significantly simplifies building, deploying, and managing your cloud resources. Organize your portal to custom-fit your work and your work style. Stay on top of the things that matter most by pinning them to your dashboard. Resize tiles to show just the right amount of detail, and share insights across apps and resources.

<https://azure.microsoft.com/en-us/get-started/azure-portal>

Use fine-grained access control

Role-based access control lets you select exactly who can manage what. Grant explicit management and access rights to the account, service, and operation levels for individuals and groups.



<https://azure.microsoft.com/en-us/get-started/azure-portal>

Create a Traffic Manager profile

Create a Traffic Manager profile that directs user traffic based on endpoint priority.

- 1. On the upper-left side of the screen, select **Create a resource**. Then search for **Traffic Manager profile** and select **Create**.
- 2. In the **Create Traffic Manager profile**, enter, or select these settings:

Expand table

Setting	Value
Name	Enter a unique name for your Traffic Manager profile.
Routing method	Select Priority .
Subscription	Select the subscription you want the traffic manager profile applied to.
Resource group	Select myResourceGroupTM1 .
Location	This setting refers to the location of the resource group. It has no effect on the Traffic Manager profile that will be deployed globally.

<https://learn.microsoft.com/en-us/azure/traffic-manager/quickstart-create-traffic-manager-profile>

Home > Create a resource > Marketplace > Traffic Manager profile

Create Traffic Manager profile

Name *

myTMprofile

trafficmanager.net

Routing method

Priority

Subscription *

Azure Subscription

Resource group *

myResourceGroupTM1

Create new

Resource group location

East US

Create

Automation options

3. Select **Create**.

<https://learn.microsoft.com/en-us/azure/traffic-manager/quickstart-create-traffic-manager-profile>

4. Enter, or select, these settings:

Expand table

Setting	Value
Type	Select Azure endpoint.
Name	Enter <i>myPrimaryEndpoint</i> .
Target resource type	Select App Service.
Target resource	Select Choose an app service > East US.
Priority	Select 1. All traffic goes to this endpoint when it's healthy.

Add endpoint

myTMprofile

Type *

Azure endpoint

Name *

myPrimaryEndpoint

Target resource type

App Service

Target resource *

myWebAppEastUS (East US)

Priority *

1

Custom Header settings

Add as disabled

Add

<https://learn.microsoft.com/en-us/azure/traffic-manager/quickstart-create-traffic-manager-profile>

6. To create a failover endpoint for your second Azure region, repeat steps 3 and 4 with these settings:

 Expand table

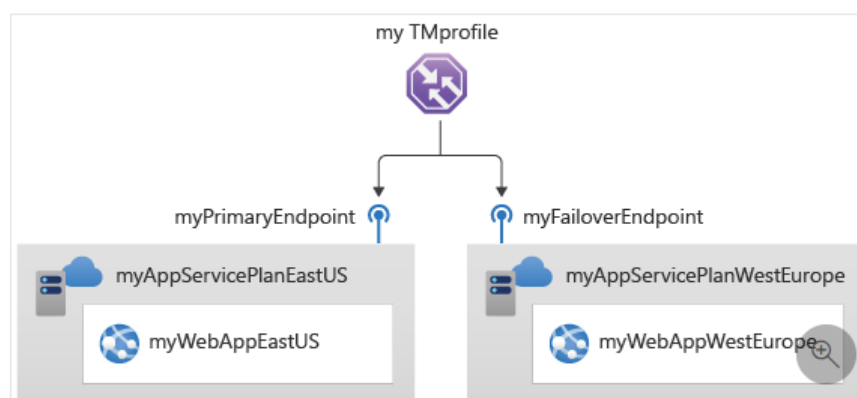
Setting	Value
Type	Select Azure endpoint .
Name	Enter <i>myFailoverEndpoint</i> .
Target resource type	Select App Service .
Target resource	Select Choose an app service > West Europe .
Priority	Select 2. All traffic goes to this failover endpoint if the primary endpoint is unhealthy.

7. Select **Add**.

When you're done adding the two endpoints, they're displayed in **Traffic Manager profile**. Notice that their monitoring status is **Online** now.

<https://learn.microsoft.com/en-us/azure/traffic-manager/quickstart-create-traffic-manager-profile>

156. Azure Front Door/CDN and Traffic Manager further provides said at least one policy to the name server network, wherein the decision tree comprises one or more resource nodes, and one or more branch nodes, wherein the one or more resource nodes specify one or more answers to be provided in response to a Domain Name Service (DNS) request, and wherein the one or more branch nodes specify one or more decision criteria to be applied, and wherein the GUI supports the specification of: (i) one or more answers for the one or more resource nodes, and (ii) one or more decision criteria for the one or more branch nodes, wherein the one or more criteria are selected from criteria related to: world zones, countries, states, time zones, and blocks of Internet Protocol (IP) addresses, and wherein the one or more resource nodes are selected from: IP addresses, canonical name (CNAME) records, mail exchange (MX) records, name server (NS) records, and load sharing server sets; and wherein the answers are selected from: IP addresses and CNAMEs.



<https://learn.microsoft.com/en-us/azure/traffic-manager/quickstart-create-traffic-manager-profile>

Create a Traffic Manager profile

Create a Traffic Manager profile that directs user traffic based on endpoint priority.

- 1. On the upper-left side of the screen, select **Create a resource**. Then search for **Traffic Manager profile** and select **Create**.
- 2. In the **Create Traffic Manager profile**, enter, or select these settings:

Expand table

Setting	Value
Name	Enter a unique name for your Traffic Manager profile.
Routing method	Select Priority .
Subscription	Select the subscription you want the traffic manager profile applied to.
Resource group	Select <i>myResourceGroupTM1</i> .
Location	This setting refers to the location of the resource group. It has no effect on the Traffic Manager profile that will be deployed globally.

<https://learn.microsoft.com/en-us/azure/traffic-manager/quickstart-create-traffic-manager-profile>

Home > Create a resource > Marketplace > Traffic Manager profile >

Create Traffic Manager profile

Name *

myTMprofile

trafficmanager.net

Routing method

Priority

Subscription *

Azure Subscription

Resource group *

myResourceGroupTM1

Create new

Resource group location

East US

Create

Automation options

- 3. Select **Create**.

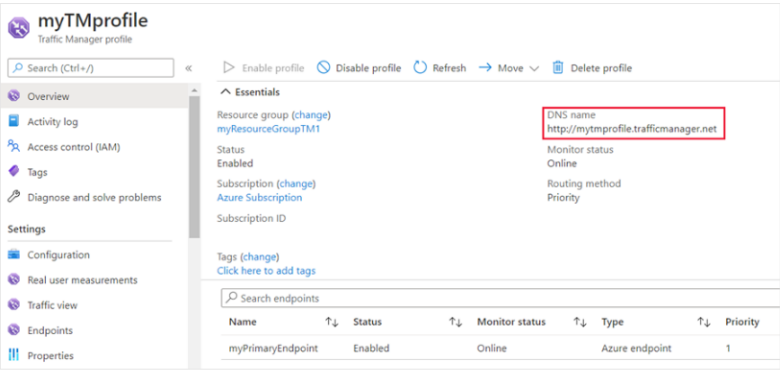
<https://learn.microsoft.com/en-us/azure/traffic-manager/quickstart-create-traffic-manager-profile>

Test Traffic Manager profile

In this section, you'll check the domain name of your Traffic Manager profile. You'll also configure the primary endpoint to be unavailable. Finally, you get to see that the web app is still available. It's because Traffic Manager sends the traffic to the failover endpoint.

Check the DNS name

1. In the portal's search bar, search for the **Traffic Manager profile** name that you created in the preceding section.
2. Select the traffic manager profile. The **Overview** appears.
3. The **Traffic Manager profile** displays the DNS name of your newly created Traffic Manager profile.



<https://learn.microsoft.com/en-us/azure/traffic-manager/quickstart-create-traffic-manager-profile>

Traffic Manager routing methods

Article • 08/08/2024 • 15 contributors

[Feedback](#)

In this article

- [Priority traffic-routing method](#)
- [Weighted traffic-routing method](#)
- [Performance traffic-routing method](#)
- [Geographic traffic-routing method](#)
- [Show 3 more](#)

Azure Traffic Manager supports six traffic-routing methods to determine how to route network traffic to the various service endpoints. For any profile, Traffic Manager applies the traffic-routing method associated to it to each DNS query it receives. The traffic-routing method determines which endpoint is returned in the DNS response.

The following traffic routing methods are available in Traffic Manager:

- **Priority:** Select **Priority** routing when you want to have a primary service endpoint for all traffic. You can provide multiple backup endpoints in case the primary or one of the backup endpoints is unavailable.
- **Weighted:** Select **Weighted** routing when you want to distribute traffic across a set of endpoints based on their weight. Set the weight the same to distribute evenly across all endpoints.
- **Performance:** Select **Performance** routing when you have endpoints in different geographic locations and you want end users to use the "closest" endpoint for the lowest network latency.
- **Geographic:** Select **Geographic** routing to direct users to specific endpoints (Azure, External, or Nested) based on where their DNS queries originate from geographically. With this routing method, it enables you to be in compliance with scenarios such as data sovereignty mandates, localization of content & user experience and measuring traffic from different regions.
- **Multivalue:** Select **Multivalue** for Traffic Manager profiles that can only have IPv4/IPv6 addresses as endpoints. When a query is received for this profile, all healthy endpoints are returned.
- **Subnet:** Select **Subnet** traffic-routing method to map sets of end-user IP address ranges to a specific endpoint. When a request is received, the endpoint returned will be the one mapped for that request's source IP address.

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

All Traffic Manager profiles have health monitoring and automatic failover of endpoints. For more information, see [Traffic Manager Endpoint Monitoring](#). Within a Traffic Manager profile, you can only configure one traffic routing method at a time. You can select a different traffic routing method for your profile at any time. Your changes will be applied within a minute without any downtime. You can combine traffic routing methods by using nested Traffic Manager profiles. Nesting profiles allows for sophisticated traffic-routing configurations that meet the needs of larger and complex applications. For more information, see [nested Traffic Manager profiles](#).

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

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: August 11, 2026

Respectfully submitted,

/s/ Ryan D. Dykal

Ryan D. Dykal (MO State Bar No. 60905)
Jordan T. Bergsten (DC Bar No. 90032105)
Mark D. Schafer (*pro hac vice* forthcoming)
Philip Eckert (*pro hac vice* forthcoming)
Anita Liu (TX State Bar No. 24134054)
BOIES SCHILLER FLEXNER LLP
1401 New York Ave, NW
Washington, DC 20005
(t): 202-274-1109
rdykal@bsfllp.com
jbergsten@bsfllp.com
mschafer@bsfllp.com
peckert@bsfllp.com
aliu@bsfllp.com

Carolyn Drell (*pro hac vice* forthcoming)
BOIES SCHILLER FLEXNER LLP
333 Main Street
Armonk, NY 10504
(t): 914-749-8200
cdrell@bsfllp.com

Counsel for Plaintiff
Sandpiper CDN, LLC