

**IN THE UNITED STATES DISTRICT COURT
FOR THE DISTRICT OF DELAWARE**

ACCELERATION BAY, LLC, a Delaware)	
Limited Liability Corporation,)	
)	
Plaintiff,)	
)	
v.)	C.A. No.
)	
AMAZON WEB SERVICES, INC., a)	DEMAND FOR JURY TRIAL
Delaware Corporation,)	
)	
Defendant.)	

COMPLAINT FOR PATENT INFRINGEMENT

Plaintiff Acceleration Bay, LLC (“Acceleration Bay”) files this Complaint for Patent Infringement and Demand for Jury Trial against Amazon Web Services, Inc. (“Amazon”) and alleges as follows:

THE PARTIES

1. Acceleration Bay is a Delaware limited liability corporation, with its principal place of business at 370 Bridge Parkway, Redwood City, California 94065.

2. Amazon Web Services, Inc. (“AWS Inc.”) is a corporation organized and existing under the laws of the State of Delaware, and upon information and belief, has its principal place of business at 410 Terry Avenue North, Seattle, Washington 98109. AWS Inc. is a wholly owned subsidiary of Amazon.com, Inc. (“Amazon.com”). AWS Inc. may be served through its agent for service of process, Corporation Service Company, 251 Little Falls Drive in Wilmington, Delaware.

JURISDICTION AND VENUE

3. This action for patent infringement arises under the patent laws of the United States, 35 U.S.C. § 101 *et seq.* This Court has original jurisdiction over this controversy pursuant to 28 U.S.C. §§ 1331 and 1338.

4. Venue is proper in this District pursuant to 28 U.S.C. §§ 1391(b) and (c) and 1400(b).

5. This Court has personal jurisdiction over Amazon because it is incorporated within this District.

6. Additionally, this Court can exercise personal jurisdiction over Amazon because it has committed acts of direct and indirect infringement in this District, including through selling infringing products and services in this District and because Acceleration Bay's claims arise out of and relate to Amazon's acts of direct and indirect infringement in this District, and because the exercise of jurisdiction by this Court over Amazon in this action would be reasonable. Amazon has more than minimal contacts with this District such that the maintenance of this action within this District would not offend traditional notions of fair play and substantial justice.

ACCELERATION BAY'S ASSERTED PATENTS

7. On March 2, 2004, the United States Patent and Trademark Office ("PTO") issued U.S. Patent No. 6,701,344 (the "'344 Patent"), entitled "Distributed Game Environment." The '344 Patent lists Fred B. Holt and Virgil E. Bourassa as its inventors. Attached hereto as Exhibit 1 is a true and correct copy of the '344 Patent.

8. All rights title, and interest in the '344 Patent have been assigned to Acceleration Bay, which is the sole owner of the '344 Patent.

9. On June 2, 2022, Acceleration Bay submitted statutory disclaimers with the PTO disclaiming Claims 1-11 and 16-19 of the '344 Patent.

10. The '344 Patent is generally directed toward systems for an effective broadcast technique using a regular network. By implementing such a broadcast technique, the system is able to provide a broadcast channel using an underlying network system that sends messages on a point-to-point basis.

11. On March 30, 2004, the PTO issued U.S. Patent No. 6,714,966 (the "'966 Patent"), entitled "Information Delivery Service." The '966 Patent lists Fred B. Holt and Virgil E. Bourassa as its inventors. Attached hereto as Exhibit 2 is a true and correct copy of the '966 Patent.

12. All rights title, and interest in the '966 Patent have been assigned to Acceleration Bay, which is the sole owner of the '966 Patent.

13. On June 2, 2022, Acceleration Bay submitted statutory disclaimers with the PTO disclaiming Claims 1-11 and 16 of the '966 Patent.

14. The '966 Patent is generally directed toward systems for providing an information delivery service using a regular network. One of the ways this is accomplished is by sending data through neighbor participants.

15. On May 4, 2004, the PTO issued U.S. Patent No. 6,732,147 (the "'147 Patent"), entitled "Leaving a Broadcast Channel." The '147 Patent lists Fred B. Holt and Virgil E. Bourassa as its inventors. Attached hereto as Exhibit 3 is a true and correct copy of the '147 Patent.

16. All rights title, and interest in the '147 Patent have been assigned to Acceleration Bay, which is the sole owner of the '147 Patent.

17. On June 2, 2022, Acceleration Bay submitted statutory disclaimers with the PTO disclaiming Claims 11, 14-16 of the '147 Patent.

18. The '147 Patent is generally directed towards methods and systems for leaving a broadcast channel. One of the ways this is accomplished is by sending messages to a second computer, so that the second computer can connect to a third computer to maintain a regular network.

19. On December 7, 2004, the PTO issued U.S. Patent No. 6,829,634 (the "'634 Patent"), entitled "Broadcasting Network." The '634 Patent lists Fred B. Holt and Virgil E. Bourassa as its inventors. Attached hereto as Exhibit 4 is a true and correct copy of the '634 Patent.

20. All rights title, and interest in the '634 Patent have been assigned to Acceleration Bay, which is the sole owner of the '634 Patent.

21. On June 2, 2022, Acceleration Bay submitted statutory disclaimers with the PTO disclaiming Claims 1-9, 19 and 22 of the '634 Patent.

22. The '634 Patent is generally directed towards methods and systems for broadcasting data across a regular network. One of the ways this is accomplished is by sending data received from neighbor participants to other neighbor participants.

23. On June 21, 2005, the PTO issued U.S. Patent No. 6,910,069 (the "'069 Patent"), entitled "Joining a Broadcast Channel." The '069 Patent lists Fred B. Holt and Virgil E. Bourassa as its inventors. Attached hereto as Exhibit 5 is a true and correct copy of the '069 Patent.

24. All rights title, and interest in the '069 Patent have been assigned to Acceleration Bay, which is the sole owner of the '069 Patent.

25. The '069 Patent is generally directed towards methods for adding a participant to a network without placing a high overhead on the underlying network.

26. The claims of the '344, '966, '147, '634, and '069 Patents (the "Asserted Patents") are directed to patent-eligible subject matter, for the reasons set forth in the Declaration of Dr. Michael Goodrich ("Goodrich Decl."), attached hereto as Exhibit 6 and incorporated by reference.

27. The Court previously found that the claims of the '344, '966, and '634 Patents cover patent eligible subject matter because they "are directed to an innovative network structure for the distribution of data as the number of participants in a computer network is scaled." Memorandum Order Denying Defendants' Motion to Dismiss, *Acceleration Bay LLC v. Take-Two Interactive Software, Inc.* ("*Take-Two*"), No. 16-cv-00455-RGA, D.I. 245, , at 6 (D. Del. Aug. 29, 2017), attached hereto as Exhibit 7. These claims "require a specific and . . . innovative structure of message-forwarding" and are thus "necessarily rooted in computer technology in order to overcome a problem specifically arising in the realm of computer networks." *Id.* at 7 (citation omitted). The claims "require a specific type of communication structure designed to improve the way computers communicate as participants scale." *Id.* (citation omitted). They "are 'specifically designed to achieve an improved technological result in conventional industry practice.'" *Id.* at 8. While not directly addressed in the Court's patent-eligibility analysis in *Take-Two* or in the Goodrich Decl., the same reasoning applies to the claims of the '147 and '069 Patents.

28. Prior to the inventions claimed in the Asserted Patents, there was a need for improved networks and methods for sharing data between widely-distributed participants. It was impractical to have every participant directly communicate because the number of connections

quickly becomes unmanageable and individual connections often fail or operate slowly. The Asserted Patents solve these problems with a very specific and inventive concept: a better network that serves as a broadcast channel for distribution of data among various participants. The broadcast channel network exists as a logical overlay to a series of point-to-point connections between participants. The broadcast channel is formed as an m-regular network, where each participant has a regular number of neighbors. Participants pass data to their neighbors in the network, who then forward the message to their neighbors, and so on, rather than being directly connected to all the participants in the network. Goodrich Decl. at ¶¶ 27-30.

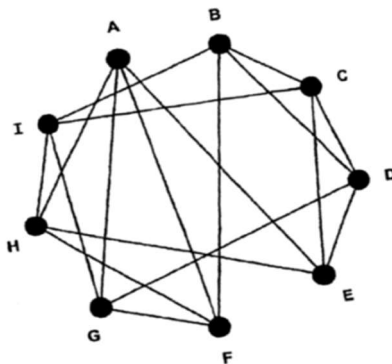
29. This non-conventional overlay network structure described in the preceding paragraph provides tangible benefits. The structure allows for rapid and reliable dissemination of data through the network because each participant forwards data to a manageable subset of network participants. For example, in a 5-regular network, a participant sends a message to its five neighbors each of which forward the message to their own neighbors, thus rapidly distributing the message over even a large network. Data will still be rapidly delivered, even if individual connections fail or operate slowly, because of the alternative pathways formed by the network, i.e. each neighbor is the start of a potential path to all other participants. This structure is also flexible, allowing for participants to be added and dropped while the network is operating. Goodrich Decl. at ¶¶ 29-35, 74-80.

30. The Broadcast Patents cover networks for a variety of applications. The '344 Patent describes “[a] [d]istributed [g]ame [e]nvironment,” which “is provided by a game application program executing on each player’s computer” ’344 Patent, 16:29-34; Goodrich Decl., ¶¶ 37-38. Each time a player takes an action, a message representing that action can be broadcast on the game’s broadcast channel. Goodrich Decl., ¶ 39. The broadcast channel allows

a player to send messages (e.g., communications or strategy information) to one or more other players. '344 Patent, 16:36-43; Goodrich Decl., ¶ 39.

31. The '966 Patent provides “an information delivery service application” which “allows participants to monitor messages as they are broadcast on the broadcast channel.” '966 Patent, 16:25-28; Goodrich Decl., ¶ 42. The '634 Patent covers “a non-routing table based method for broadcasting messages in a network,” where, rather than have each participant store routing-table information (i.e., information about non-neighboring participants in the network), each participant in the network implements the broadcast using information about neighboring participants. '634 Patent, 2:45-67; Goodrich Decl., ¶ 45.

32. Rather than forming the network as a complete graph, which could cause congestion by requiring each participant to be directly connected to all other participants, requiring a large number of connections, (e.g., '344 Patent, 1:44-57), the participants are connected so as to form an m-regular, incomplete network. Goodrich Decl., ¶ 28. This topology ensures reliability and scalability of the network. *Id.* In one example, the '344 Patent shows “a graph that is 4-regular and 4-connected which represents the broadcast channel,” meaning that each participant is connected to four neighbors and at least four connections must be broken to prevent a participant from being able to communicate, at least indirectly, with all other participants. '344 Patent, 4:48-49, Fig. 1 (below).



33. Network protocols are typically modeled conceptually as being partitioned into layers, which collectively are called the network protocol stack. Goodrich Decl., ¶ 18. Each layer provides a set of services and functionality guarantees for higher layers and, to the extent possible, each layer does not depend on details or services from higher levels. *Id.* To reduce complexity, most networks are designed with a small number of layers, from the physical layer, at the bottom, where computer hardware interfaces with copper wire or wireless radio, to the application layer, at the top, where the user interacts with the software. *Id.*

34. The broadcast channel is implemented through a “graph of point-to-point connections” that “overlays the underlying network.” *E.g.*, ’344 Patent, 4:23-26. Importantly, the claimed networks establish a gaming or information delivery environment at the ***application level***, rather than the underlying transport or network levels. *E.g.*, ’344 Patent, 4:15-19 (“The logical broadcast channel is implemented using an underlying network system (e.g., the Internet) that allows each computer connected to the underlying network system to send messages to each other connected computer using each computer’s address.”); Goodrich Decl., ¶ 30.

35. Further emphasizing that the networks are implemented at the application level, the ’344 Patent explains that “[t]he broadcast channel is well suited for computer processes (e.g., ***application programs***) that execute collaboratively, such as network meeting programs.” ’344 Patent, 15:17-20 (emphasis added). The patent further explains that “[t]he ***application program*** invokes the connect component to establish a connection to a designated broadcast channel.” *Id.* at 16:9-11 (emphasis added). Moreover, “[t]he broadcast component is invoked by the ***application program*** to broadcast messages in the broadcast channel.” *Id.* at 16:26-28 (emphasis added). With respect to the game environment, the patent explains that “[t]he game environment is provided by a game ***application program*** executing on each player’s computer

that interacts with a broadcaster component.” *Id.* at 16:31-34 (emphasis added), 15:29-49; Goodrich Decl., ¶ 38.

OVERVIEW OF AMAZON’S PRODUCTS AND SERVICES

36. Amazon offers a variety of products and services for its customers to use to provide Internet-based services, including broadcasting game data for online multiplayer games and for providing content delivery.

37. Amazon also uses the products and services for its gaming and content delivery services operated by its parent Amazon.com, such as Amazon Prime Video, Amazon Luna, and Twitch.

38. The following is an overview of these products and services that give rise to Amazon’s infringement of the Asserted Patents, as discussed in the subsequent sections of this complaint.

AMAZON’S AMAZON WEB SERVICE

39. Amazon Web Service (“AWS”) is a scalable cloud infrastructure that includes “multiple Availability Zones connected by low latency, high throughput, and highly redundant networking. AWS has 84 Availability Zones within 26 geographic regions around the world.” Ex. 8, <https://aws.amazon.com/what-is-aws/>.

40. According to Amazon, AWS “is the world’s most comprehensive and broadly adopted cloud platform, offering over 200 fully featured services from data centers globally. Millions of customers—including the fastest-growing startups, largest enterprises, and leading government agencies—are using AWS to lower costs, become more agile, and innovate faster.” *Id.*

41. AWS supports overlay networks. For example, Amazon publishes instructions to deploy a “Cohesive Networks VNS3 overlay network on the AWS Cloud in about 15 minutes, following best practices from AWS and Cohesive Networks” using AWS technologies, such as Availability Zones, Virtual Private Cloud, and Elastic Compute Cloud instances using Auto Scaling. Ex. 9, <https://aws.amazon.com/quickstart/architecture/cohesive-networks-vns3/>.

AMAZON’S ELASTIC KUBERNETES SERVICE

42. AWS includes Amazon Elastic Kubernetes Service (“EKS”), which is “a managed service that [customers] can use to run Kubernetes on AWS without needing to install, operate, and maintain [their] own Kubernetes control plane or nodes. Kubernetes is an open-source system for automating the deployment, scaling, and management of containerized applications.” Ex. 10 at 1, <https://docs.aws.amazon.com/eks/latest/userguide/eks-ug.pdf>.

Amazon EKS “[r]uns and scales the Kubernetes control plane across multiple AWS Availability Zones to ensure high availability,” “[a]utomatically scales control plane instances based on load, detects and replaces unhealthy control plane instances,” “[i]s integrated with many AWS services to provide scalability and security,” including Amazon ECR for container images, Elastic Load Balancing for load distribution, IAM for authentication, and Amazon VPC for isolation. *Id.*

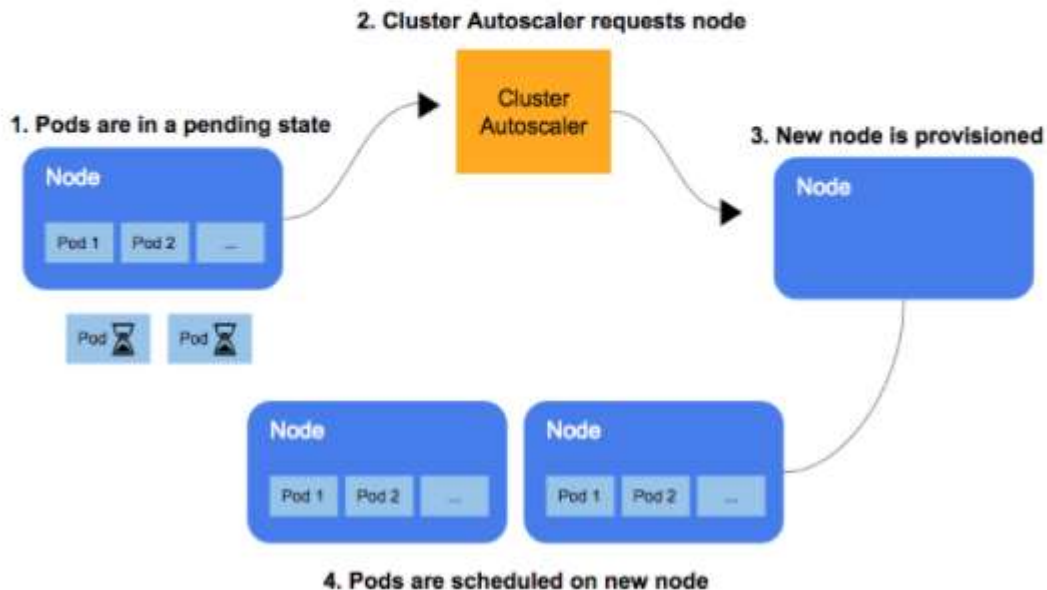
43. Amazon EKS utilizes a control plane and data plane architecture, of which the control plane “consists of at least two API server instances and three etcd instances that run across three Availability Zones within an AWS Region.” Ex. 10 at 1, <https://docs.aws.amazon.com/eks/latest/userguide/eks-ug.pdf>.

44. Amazon EKS “[a]ctively monitors the load on control plane instances and automatically scales them to ensure high performance,” “[a]utomatically detects and replaces

unhealthy control plane instances, restarting them across the Availability Zones within the AWS Region as needed,” and “[l]everages the architecture of AWS Regions in order to maintain high availability.” *Id.* This feature is called Cluster Autoscaling.

When Does The Cluster Autoscaler Add Capacity?

The cluster autoscaler increases the size of the cluster when there are pods that are not able to be scheduled due to resource shortages. It can be configured to not scale up or down past a certain number of machines. Here is an overview of how it makes scaling decisions:



Ex. 11 at 2-3, <https://medium.com/kubecost/understanding-kubernetes-cluster-autoscaling-675099a1db92>.

AMAZON’S ELASTIC CLOUD COMPUTING

45. Amazon’s Elastic Cloud Computing (“EC2”) “is a compute web service that offers secure, resizable compute capacity in the cloud. It is designed for scalable deployments and optimizing [customers’] workloads.” Ex. 12 at 2,

<https://aws.amazon.com/free/compute/lightsail-vs-ec2/>. Using Amazon’s EC2, users can “launch as many or as few virtual servers as [they] need, configure security and networking, and manage storage” and “enables [customers] to scale up or down to handle changes in requirements or spikes in popularity, reducing [their] need to forecast traffic.” Ex. 13 at 1, <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-ug.pdf>.

AMAZON’S AWS APP MESH

46. Amazon’s AWS App Mesh (“App Mesh”) “is a service mesh that provides application-level networking to make it easy for [customers’] services to communicate with each other across multiple types of compute infrastructure. App Mesh gives end-to-end visibility and high-availability for [customers’] applications.” Ex. 14 at 1, <https://aws.amazon.com/app-mesh>. App Mesh operates “with AWS Fargate, Amazon EC2, Amazon ECS, Amazon EKS, and Kubernetes running on AWS, to better run [customers’] application at scale. App Mesh also integrates with AWS Outposts for [customers’] applications running on-premises.” *Id.* at 2. App Mesh is based on the Envoy proxy technology. Ex. 15 at 137, <https://docs.aws.amazon.com/app-mesh/latest/userguide/app-mesh-ug.pdf>.

47. A service mesh is an infrastructure layer dedicated to handling service-to-service communication, usually through an array of lightweight network proxies deployed alongside the application code. *Id.* at 1. A service mesh acts as “a logical boundary for network traffic between the services that reside within it. After [customers] create [a] service mesh, [they] can create virtual services, virtual nodes, virtual routers, and routes to distribute traffic between the applications in [their] mesh.” *Id.* at 57.

AMAZON'S LAMBDA

48. Amazon's Lambda "is a serverless, event-driven compute service that lets [customers] run code for virtually any type of application or backend service without provisioning or managing servers. [Customers] can trigger Lambda from over 200 AWS services and software as a service (SaaS) applications, and only pay for what [they] use." Ex. 16 at 2, <https://aws.amazon.com/lambda/>.

49. Lambda supports built-in fault tolerance and "maintains compute capacity across multiple Availability Zones (AZs) in each AWS Region to help protect . . . code against individual machine or data center facility failures. Both . . . Lambda and the functions running on the service deliver predictable and reliable operational performance. . . . Lambda is designed to provide high availability for both the service itself and the functions it operates." Ex. 17 at 3, <https://aws.amazon.com/lambda/features/?pg=ln&sec=hs>.

50. Lambda "automatically scales to support the rate of incoming requests without any manual configuration. There is no limit to the number of requests [customers'] code can handle. . . . Lambda typically starts running . . . code within milliseconds of an event. Since Lambda scales automatically, the performance remains consistently high as the event frequency increases. Since . . . code is stateless, Lambda can start as many instances as needed without lengthy deployment and configuration delays." *Id.* at 4.

AMAZON'S CLOUDFRONT (LAMBDA@EDGE)

51. Amazon's CloudFront "is a content delivery network (CDN) service built for high performance, security, and developer convenience." Ex. 18 at 2, <https://aws.amazon.com/cloudfront/>. CloudFront, also known as Lambda@Edge, uses "Node.js and Python Lambda functions to customize content that CloudFront delivers, executing the

functions in AWS locations closer to the viewer. The functions run in response to CloudFront events, without provisioning or managing servers.” Ex. 19 at 535,

<https://docs.aws.amazon.com/lambda/latest/dg/lambda-dg.pdf>; Ex. 20,

https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/AmazonCloudFront_DeveloperGuide.pdf.

52. Amazon CloudFront “peers with thousands of Tier 1/2/3 telecom carriers globally, is well connected with all major access networks for optimal performance, and has hundreds of terabits of deployed capacity. CloudFront Edge locations are connected to the AWS Regions through the AWS network backbone - fully redundant, multiple 100GbE parallel fiber that circles the globe and links with tens of thousands of networks for improved origin fetches and dynamic content acceleration.” Ex. 21 at 1-2,

<https://aws.amazon.com/cloudfront/features/?whats-new-cloudfront>. Amazon CloudFront

includes a global network of “310+ Points of Presence (300+ Edge locations and 13 regional mid-tier caches) in 90+ cities across 47 countries.” *Id.* at 2.

AMAZON’S VIRTUAL PRIVATE CLOUD PEERING

53. Amazon’s Virtual Private Cloud Peering (“VPC Peering”) “enables [customers] to launch AWS resources into a virtual network that [they have] defined.” Ex. 22 at 1,

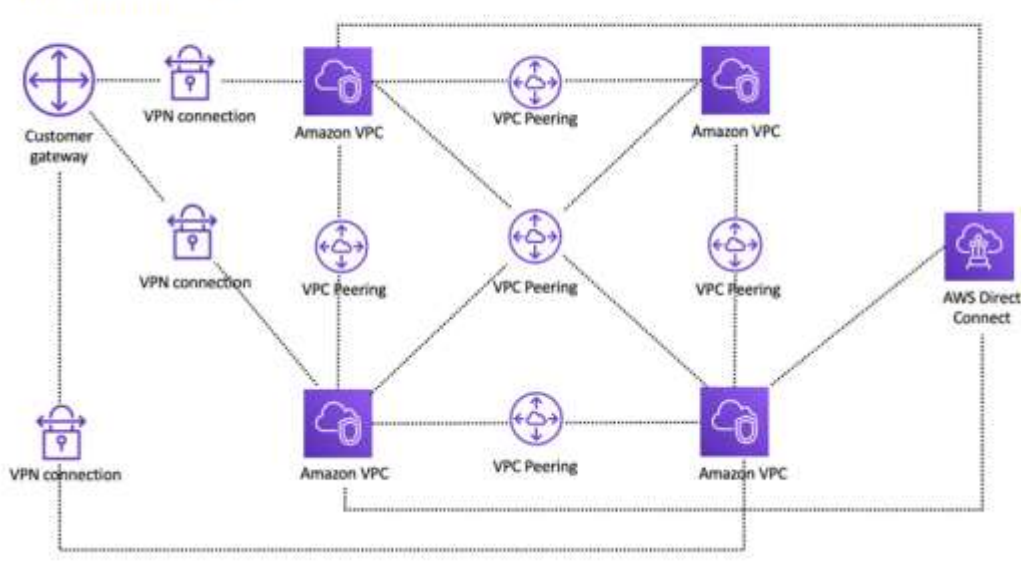
<https://docs.aws.amazon.com/vpc/latest/peering/vpc-pg.pdf>. VPC Peering is a “networking connection between two VPCs that enables [routing] traffic between them using private IPv4 addresses or IPv6 addresses. Instances in either VPC can communicate with each other as if they are within the same network. [Customers] can create a VPC peering connection between [their] own VPCs, or with a VPC in another AWS account. The VPCs can be in different regions (also known as an inter-region VPC peering connection).” *Id.*

54. VPC peering occurs across “different AWS Regions (also called inter-Region VPC peering). This allows VPC resources including EC2 instances, Amazon RDS databases and Lambda functions that run in different AWS Regions to communicate with each other using private IP addresses, without requiring gateways, VPN connections, or separate network appliances Traffic always stays on the global AWS backbone, and never traverses the public internet, which reduces threats, such as common exploits, and DDoS attacks.” *Id.*

55. VPC peering results in “point-to-point connectivity” of VPCs using “full bidirectional connectivity between the VPCs. This peering connection is used to route traffic between the VPCs. VPCs across accounts and AWS Regions can also be peered together.” Ex. 23 at 4, <https://docs.aws.amazon.com/whitepapers/latest/building-scalable-secure-multi-vpc-network-infrastructure/building-scalable-secure-multi-vpc-network-infrastructure.pdf#welcome>.

56. Interconnecting multiple VPCs (e.g., tens to hundreds) at scale “results in a mesh of hundreds to thousands of peering connections.” For example, 100 VPCs in a full mesh peering configuration would require “4,950 peering connections $[n(n-1)/2]$ where n =total number of VPCs.” *Id.* However, the VPCs are limited to 125 connection, making the network incomplete.

At scale, when you have tens to hundreds of VPCs, interconnecting them with peering results in a mesh of hundreds to thousands of peering connections, which are difficult to manage and scale. For example, if you have 100 VPCs and you want to setup a full mesh peering between them, it will take 4,950 peering connections $[n(n-1)/2]$ where n =total number of VPCs. There is a **maximum limit** of up to 125 active peering connections per VPC.



Id.

AMAZON’S ELASTIC LOAD BALANCING

57. Amazon’s Elastic Load Balancing (“ELB”) “automatically distributes . . . incoming traffic across multiple targets, such as EC2 instances, containers, and IP addresses, in one or more Availability Zones. It monitors the health of its registered targets, and routes traffic only to the healthy targets. Elastic Load Balancing scales [customers’] load balancer as [their] incoming traffic changes over time. It can automatically scale to the vast majority of workloads. Elastic Load Balancing supports the following load balancers: Application Load Balancers, Network Load Balancers, Gateway Load Balancers, and Classic Load Balancers. [Customers] can select the type of load balancer that best suits [their] needs.” Ex. 24 at 1, <https://docs.aws.amazon.com/elasticloadbalancing/latest/network/elb-ng.pdf>.

58. ELB operates with Availability Zones to balance network traffic among multiple nodes. ELB supports creation of “a load balancer node in the Availability Zone. By default,

each load balancer node distributes traffic across the registered targets in its Availability Zone only. If [customers] enable cross-zone load balancing, each load balancer node distributes traffic across the registered targets in all enabled Availability Zones.” *Id.*

AMAZON’S LUMBERYARD AND O3DE

59. Amazon’s Lumberyard is a game engine that “allows [customers] to create high-quality games, connect [their] games to the compute and storage of the AWS Cloud, and engage fans on Twitch.” Ex. 25 at 1, <https://docs.aws.amazon.com/lumberyard/latest/userguide/lumberyard-ug.pdf>. Amazon Open 3d Engine (“O3DE”) is the successor to Lumberyard.

60. Amazon Lumberyard and O3DE both support P2P (Peer-to-Peer) configurations. *Id.* at 2011-12.

61. Amazon Lumberyard includes GridMate, which is a “robust and flexible networking solution designed for efficient bandwidth usage and low-latency communications. [Customers] can synchronize objects over the network with GridMate’s replica framework. GridMate’s session management can be integrated with major online console services and helps . . . handle peer-to-peer and client–server topologies with host migration.” *Id.* at 2-3.

62. Amazon GridMate facilitates communications for Lumberyard, including ordered/unordered messaging, reliable/unreliable messaging, fragmenting messages, encryption, and channels on which to divide different types of traffic. Ex. 27, <https://aws.amazon.com/blogs/gametech/building-battle-tested-network-transport/>.

63. Amazon Lumberyard includes a messaging system called EBus (“Event Bus”), which is used as a broadcast communication channel between different systems. Ex. 28 at 7-8, <https://docs.aws.amazon.com/lumberyard/latest/welcomeguide/lumberyard-wg.pdf>. Amazon

Lumberyard supports “new levels of interactivity between Twitch streamers and viewers with Twitch ChatPlay.” *Id.* at 3. EventBus gives “Twitch streamers dynamic real-time broadcast customization options with Twitch Metastream [and enables] direct engagement between Twitch streamers and viewers with Twitch JoinIn.” *Id.*

AMAZON’S GAMELIFT

64. Amazon GameLift is a game hosting solution that operates on Amazon AWS for session-based, multiplayer games. Ex. 26 at 1, <https://docs.aws.amazon.com/gamelift/latest/developerguide/gamelift-dg.pdf>. Amazon GameLift “helps deliver high-performance, high-reliability, low-cost game servers while dynamically scaling [customers’] resource usage to meet worldwide player demand.” *Id.*

65. Amazon GameLift operates on Amazon EC2 resources, called instances, to deploy game servers, as well as supporting VPC peering. *Id.* at 55, 106-09. Amazon GameLift supports FlexMatch, which is a matchmaking service for multiplayer games that operate on peer-to-peer services. *Id.* at 280-81.

66. Amazon GameLift fleets support autoscaling to increase and decrease server instances based on game activity. *Id.* at 132-37.

AMAZON’S LUNA AND TWITCH

67. Amazon Luna is a cloud gaming platform developed and operated by Amazon.com. Amazon Luna allows subscribers to play a variety of videogames on multiples devices. Amazon Luna operates on AWS. Ex. 29, <https://techcrunch.com/2020/09/24/amazon-announces-luna-game-streaming-platform/> (“[t]he whole service is powered by AWS”).

68. Amazon Twitch is “a live content streaming video-platform that allows gamers to stream their games for others to watch while they play.” Ex. 30,

<https://rohan6820.medium.com/aws-case-study-twitch-324ecf8288aa>. AWS hosts and delivers content for Twitch, and uses other Amazon services, such as CloudFront, EC2, Elastic Load Balancer, Rekognition, and S3. *Id.*

69. Amazon Twitch includes Twitch software applications that operates on Windows, Android and iOS platforms. Ex. 31, <https://www.twitch.tv/downloads>.

70. Amazon Luna integrates with Amazon Twitch, which can be used to broadcast directly to Amazon Twitch from Amazon Luna using AWS. Ex. 38, <https://www.amazon.com/gp/help/customer/display.html?nodeId=GEZHHTM8ZRSZRQQC> (“Broadcast Directly to Twitch Using Amazon Luna.”).

AMAZON’S AMAZON PRIME VIDEO

71. Amazon Prime Video is Amazon.com’s video delivery streaming service which operates on Amazon AWS. Ex. 32 at 2, <https://aws.amazon.com/solutions/case-studies/amazon-prime-video/> (“Amazon Prime Video uses the Amazon Web Service (AWS) Cloud as the underlying technology for all its services.”). Amazon Prime Video also utilizes AWS Elemental, “an Amazon Web Services company that combines deep video expertise with the power and scale of the cloud to empower media companies to deliver premium video experiences to consumers,” for live encoding, packaging, and ad insertion for NFL Football streaming. *Id.*

72. Amazon Prime Video also utilizes AWS Elemental MediaTailor, “which is used to monetize video content through the insertion of targeted advertising,” and Amazon DynamoDB, “which Amazon Prime Video uses as the key value store for the streaming platform” for high scalability and performance to support “more than 300,000 video clients.” *Id.* at 2-3.

73. Amazon Prime Video is based on additional AWS services, such as “Amazon CloudFront for delivering video with low latency to a global viewership, and Amazon Elastic Compute Cloud (Amazon EC2) for managing compute capacity.” *Id.* at 3.

74. Amazon Prime Video streams to clients via the Amazon Prime Video app, which supports “more than 600 types of TVs, mobile devices, game consoles, set-top boxes, and connected devices.” *Id.* Amazon Prime Video also “recently launched CBS All Access, which gives Amazon Prime Video subscribers access to hundreds of live local channels across the United States.” *Id.* at 4.

AMAZON’S INFRINGEMENT OF ACCELERATION BAY’S PATENTS

75. Amazon has been and is now infringing, and will continue to infringe, the Asserted Patents in this Judicial District and elsewhere in the United States by, among other things, making, using, importing, selling, and offering for sale AWS alone or in combination with Amazon Lumberyard, O3DE, Amazon Luna, Amazon Prime, Amazon Twitch, GameLift, Gridmate, EC2, EKS, CloudFront, VPC Peering, App Mesh, and Lambda (the “Accused Products”).

76. Amazon directly infringes the Asserted Patents pursuant to 35 U.S.C. § 271(a), either literally or under the doctrine of equivalents, or both.

77. Amazon also indirectly infringes the Asserted Patents by encouraging, instructing, directing, and requiring others, including its customers, purchasers, users, and developers, to perform the steps of the method claims of the Asserted Patents and to make and use the systems of the system claims of the Asserted Patents, either literally or under the doctrine of equivalents, or both.

78. Amazon is aware of the Asserted Patents and knew or was willfully blind to its infringement of the Asserted Patents. Despite this knowledge or willful blindness, Amazon has acted with blatant and egregious disregard for Acceleration Bay's patent rights with an objectively high likelihood of infringement.

79. On March 13, 2019, Acceleration Bay sent a letter to Amazon notifying it that at least Amazon and AWS' CloudFront were and are infringing Acceleration Bay's patents, including the '966 Patent, '147 Patent, '634 Patent, '069 Patent, and '497 Patent.

80. In addition to notifying Amazon of its infringement, Acceleration Bay offered to negotiate a license with Amazon for its use of the Asserted Patents.

81. Amazon did not address Acceleration Bay's allegations of infringement and did not respond to Acceleration Bay's proposal to negotiate a license to the Asserted Patents.

82. Despite Amazon's knowledge of the Asserted Patents and of its infringement, Amazon continued to infringe the Asserted Patents.

83. Amazon has not identified to Acceleration Bay any efforts that it has taken to avoid infringement of the Asserted Patents and Acceleration Bay has not seen any evidence of any attempts by Amazon to lessen its infringement. Therefore, on information and belief, Amazon has undertaken no efforts to avoid infringement of the Asserted Patents, despite Amazon's knowledge and understanding that its products and services infringe these patents.

84. Thus, Amazon's infringement of the Asserted Patents is willful and egregious, warranting enhancement of damages.

AMAZON'S ACCUSED NETWORKS ARE M-REGULAR AND INCOMPLETE

85. Amazon's AWS include networks which are m-regular and incomplete.

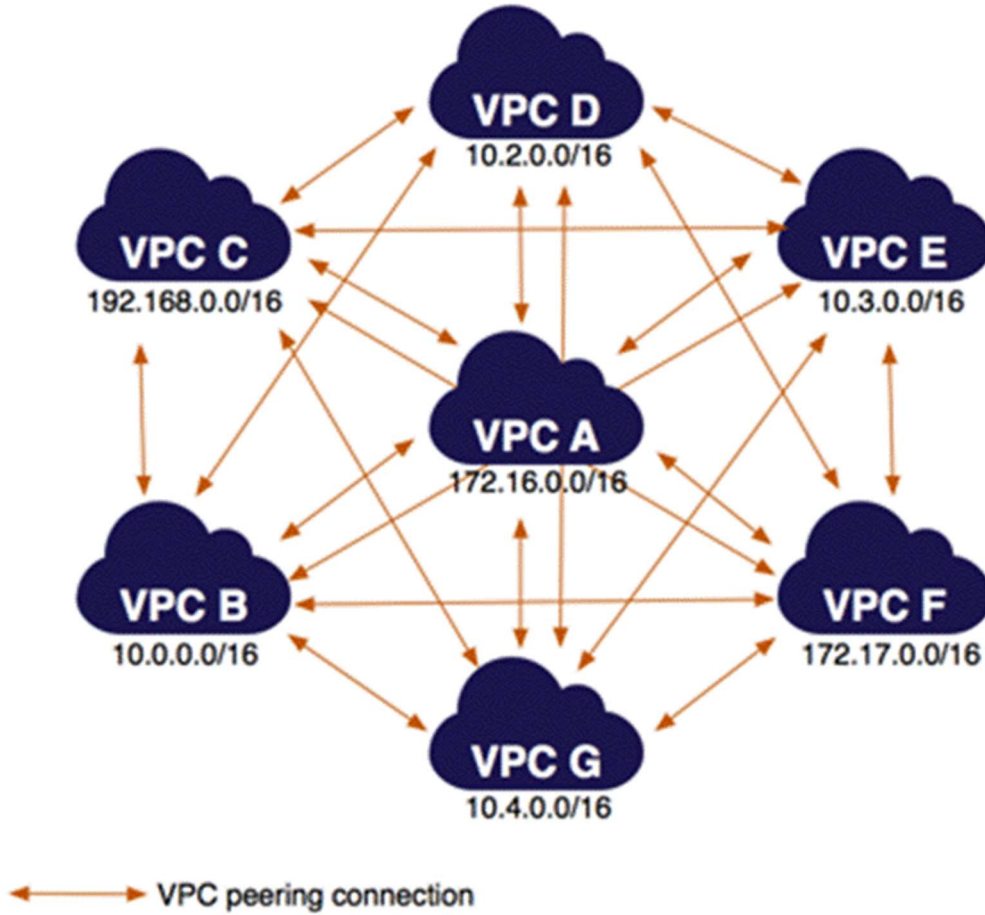
86. Acceleration Bay tested AWS and observed that AWS networks are m-regular and incomplete.

87. AWS publishes its current IP address ranges, in the form of a JSON file, which Acceleration Bay used in its testing to collect data. Ex. 33 at 932, <https://docs.aws.amazon.com/general/latest/gr/aws-general.pdf>.

88. Attached as Exhibit 39 is a visual graph of the AWS network operating on CloudFront, which shows a large set of m -regular networks where m is between 3 and 9, with each separate network distinguishable by the color corresponding to the m value for that network. Acceleration Bay collected this data by mapping the connections between thousands of nodes in the AWS network.

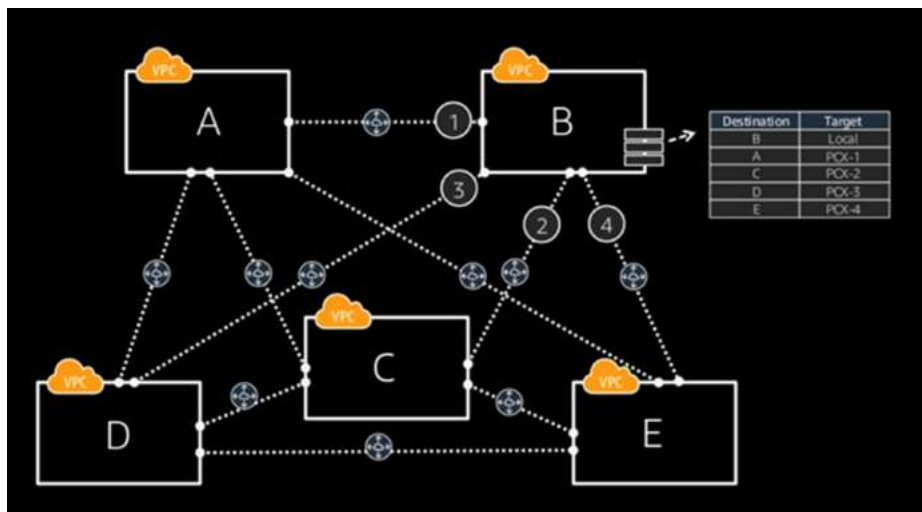
89. The AWS network is incomplete because not all peers are connected, as reflected in the data collected by Acceleration Bay and as shown in Exhibit 39.

90. The AWS network uses peer-to-peer connections to connect the participating nodes in a network. Shown below is an example of an m -regular peer-to-peer network using VPC peering. This VPC peering network overlays an underlying TCP/IP network. The depicted network is m -regular but not incomplete. On information and belief and based on the experimental data collected above, the actual VPC peering networks are both incomplete and m -regular.



Ex. 22 at 31, <https://docs.aws.amazon.com/vpc/latest/peering/vpc-pg.pdf>.

91. Shown below is an m-regular, peer-to-peer network using VPC peering. This VPC peering network overlays an underlying TCP/IP network.



Ex. 34 at Slide 42, <https://www.slideshare.net/AmazonWebServices/aws-networking-fundamentals-175148608>.

COUNT I
(Direct Infringement of the '344 Patent)

92. Acceleration Bay repeats, realleges, and incorporates by reference, as if fully set forth herein, the allegations of the preceding paragraphs, as set forth above.

93. Amazon has infringed and continues to infringe at least Claim 21 of the '344 Patent in violation of 35 U.S.C. § 271(a).

94. Amazon's infringement is based upon literal infringement or infringement under the doctrine of equivalents, or both.

95. Amazon's acts of making, using, importing, selling, and offering for sale the Accused Products has been without the permission, consent, authorization, or license of Acceleration Bay.

96. To the extent specific components are provided by or steps performed by Amazon's customers or partners, Amazon directs and controls the systems and methods of the claims and obtains benefits from the control of the system as a whole, such that those components are put into use by Amazon and the steps are deemed to be performed by Amazon. Moreover, Amazon is the final assembler of the infringing system because it controls, assembles, and combines the components to form the infringing system.

97. The Accused Products include a computer network for providing a game environment for a plurality of gaming participants, each gaming participant having connections to at least three neighbor gaming participants. Amazon Lumberyard, O3DE, GameLift, and Amazon Twitch operating on AWS provide a game environment for multiplayer games with up to 200 participants. Ex. 26 at 1-2, <https://docs.aws.amazon.com/gamelift/latest/>

[developerguide/gamelift-dg.pdf](#). Amazon Lumberyard and O3DE include GridMate, which facilitates communications for Lumberyard, including ordered/unordered messaging, reliable/unreliable messaging, fragmenting messages, encryption, and channels on which to divide different types of traffic. Ex. 27, <https://aws.amazon.com/blogs/gametech/building-battle-tested-network-transport/>. Amazon also uses AWS to provide its Amazon Luna service and its Amazon Luna over Amazon Twitch service.

98. The Accused Products send gaming data from an originating gaming participant to the other gaming participants by sending the gaming data through each of its connections to its neighbor gaming participants and wherein each gaming participant sends gaming data that it receives from a neighbor gaming participant to its other neighbor gaming participants. For example, Amazon GridMate facilitates communications for Lumberyard, including ordered/unordered messaging, reliable/unreliable messaging, fragmenting messages, encryption, and channels on which to divide different types of traffic. *Id.* Additionally, Amazon Lumberyard includes a messaging system called EBus (“Event Bus”), which is used as a broadcast communication channel between different systems. Ex. 28 at 7-8, <https://docs.aws.amazon.com/lumberyard/latest/welcomeguide/lumberyard-wg.pdf>.

99. As discussed above, the Accused Products use m -regular networks, where m is the exact number of neighbor gaming participants of each gaming participant and the number of gaming participants is at least two greater than m thus resulting in a non-complete graph. The connections between the gaming participants are peer-to-peer connections.

100. The Accused Products’ network is formed through a broadcast channel that overlays an underlying network. For example, Accused Products use the `broadcastGroupMembershipUpdate()` function to form the network, which “[t]riggers delivery of

an updated group membership list to player group. Specify which membership to broadcast (groupIdToBroadcast) and the group to receive the update (targetGroupId).” Ex. 26 at 225-26, <https://docs.aws.amazon.com/gamelift/latest/developerguide/gamelift-dg.pdf>.

101. The Accused Products’ game environment is provided by at least one game application program executing on each computer of the computer network that interacts with the broadcast channel and gaming participants can join and leave the network using the broadcast channel. For example, Amazon’s “Twitch Plays Pokémon” is a game application program which is based on Amazon Lumberyard and executes on each computer of the computer network. Ex. 35, <https://arstechnica.com/gaming/2016/02/amazon-games-launches-free-game-engine-with-native-twitch-integration/>. Amazon Lumberyard includes a “Twitch JoinIn feature that will allow for multiplayer games where Twitch broadcasters can instantly invite viewers to join them side-by-side in a session.” *Id.*

102. Additionally, Amazon GameLift “replaces the work required [for customers] to host . . . custom game servers, including buying and setting up hardware, and managing ongoing activity, security, storage, and performance tracking. Auto-scaling capabilities provide additional protection from having to pay for more resources than [the customers] need, while making sure [they] always have games available for new players to join with minimal waiting.” Ex. 26 at 1-2, <https://docs.aws.amazon.com/gamelift/latest/developerguide/gamelift-dg.pdf>.

103. Amazon’s infringement of the ’344 Patent has injured and continues to injure Acceleration Bay in an amount to be proven at trial, but not less than a reasonable royalty. Amazon’s actions are willful, blatant and in egregious disregard for Acceleration Bay’s patent rights. Amazon’s infringement has caused and is continuing to cause damage and irreparable

injury to Acceleration Bay, and Acceleration Bay will continue to suffer damage and irreparable injury unless and until that infringement is enjoined by this Court.

104. Amazon acted recklessly, willfully, wantonly, and deliberately engaged in acts of infringement of the '344 Patent, justifying an award to Acceleration Bay of increased damages under 35 U.S.C. § 284, and attorney's fees and costs incurred under 35 U.S.C. § 285.

COUNT II
(Indirect Infringement of the '344 Patent)

105. Acceleration Bay repeats, realleges, and incorporates by reference, as if fully set forth herein, the allegations of the preceding paragraphs.

106. As discussed above, Amazon knew about Acceleration Bay's patents, including the '344 Patent, and was informed it was infringing, by no later than March 13, 2019, and further knows about its infringement of the '344 Patent from its receipt of this Complaint.

107. In addition to directly infringing the '344 Patent, as discussed above with respect to Count I, Amazon knew or was willfully blind to the fact that it was inducing infringement of the '344 Patent under 35 U.S.C. § 271(b) by instructing, encouraging, directing, and requiring third parties, including its customers, to directly infringe by using and installing the Accused Products in the United States.

108. Amazon contributorily infringes the '344 Patent pursuant to 35 U.S.C. § 271(c) because it has provided software, that acts as a material component of the claimed inventions of the '344 Patent. In particular, Amazon knows that its products are particularly suited to be used in an infringing manner. Amazon has known or was willfully blind to the fact that it is contributing to the infringement of one or more claims of the '344 Patent. Amazon's software is not a staple article or commodity of commerce suitable for substantial noninfringing use.

109. Amazon knowingly and actively aided and abetted, encouraged, and contributed to the direct infringement of the '344 Patent by instructing and encouraging its customers, purchasers, users, developers, vendors, partners, and manufacturers to meet the elements of the '344 Patent with the Accused Products, as described above. Such instructions and encouragement included, but is not limited to, advising third parties to use the Accused Products in an infringing manner through direct communications, training and support materials, and customer support regarding how to install and configure the Accused Products, by advertising and promoting the use of the Accused Products in an infringing manner, and distributing release notes, webinars, guidelines, videos, manuals, white papers, and trainings to third parties on how the Accused Products must be used. *See, e.g.*, Ex. 10 (EKS User Guide) (including instructions to set up and configure EKS); Ex. 9 (Step-by-step guide to “[d]eploy an overlay network for regulated and sensitive workloads on AWS.”); Ex. 13 (AWS EC2 User Guide) (including instructions to set up and configure EC2); Ex. 15 (AWS App Mesh User Guide) (including instructions to set up and configure App Mesh); Ex. 16 (AWS Lambda website); Ex. 19 (AWS Lambda Developer Guide) (including instructions to set up and configure Lambda); Ex. 20 (Amazon CloudFront Developer Guide) (including instructions to set up and configure CloudFront); Ex. 22 (VPC Peering guide) (including instructions to set up and configure VPC Peering); Ex. 23 (AWS Whitepaper on “Building a Scalable and Secure Multi-VPC AWS Network Infrastructure”); Ex. 24 (ELB guide) (including instructions to set up and configure Elastic Load Balancing); Ex. 25 (Lumberyard User Guide) (including instructions to set up and configure Lumberyard); Ex. 27 (AWS Website) (including instructions to set up and configure GridMate); Ex. 26 (Amazon GameLift Developer Guide) (including instructions to set up and configure GameLift); Ex. 33 (AWS General Reference) (including instructions to set up and

configure AWS); Ex. 36 (EC2 Auto Scaling User Guide) (including instructions to set up and configure Auto Scaling).

110. Amazon's indirect infringement of the '344 Patent has injured and continues to injure Acceleration Bay in an amount to be proven at trial, but not less than a reasonable royalty. Amazon's actions are willful, blatant, and in egregious disregard for Acceleration Bay's patent rights. Amazon's indirect infringement has caused and is continuing to cause damage and irreparable injury to Acceleration Bay, and Acceleration Bay will continue to suffer damage and irreparable injury unless and until that infringement is enjoined by this Court.

111. Amazon acted recklessly, willfully, wantonly, and deliberately engaged in acts of indirect infringement of the '344 Patent, justifying an award to Acceleration Bay of increased damages under 35 U.S.C. § 284, and attorney's fees and costs incurred under 35 U.S.C. § 285.

COUNT III
(Direct Infringement of the '966 Patent)

112. Acceleration Bay repeats, realleges, and incorporates by reference, as if fully set forth herein, the allegations of the preceding paragraphs, as set forth above.

113. Amazon has infringed and continues to infringe at least Claim 13 of the '966 Patent in violation of 35 U.S.C. § 271(a).

114. Amazon's infringement is based upon literal infringement or infringement under the doctrine of equivalents, or both.

115. Amazon's acts of making, using, importing, selling, and offering for sale the Accused Products has been without the permission, consent, authorization, or license of Acceleration Bay.

116. To the extent specific components are provided by or steps performed by Amazon's customers or partners, Amazon directs and controls the systems and methods of the

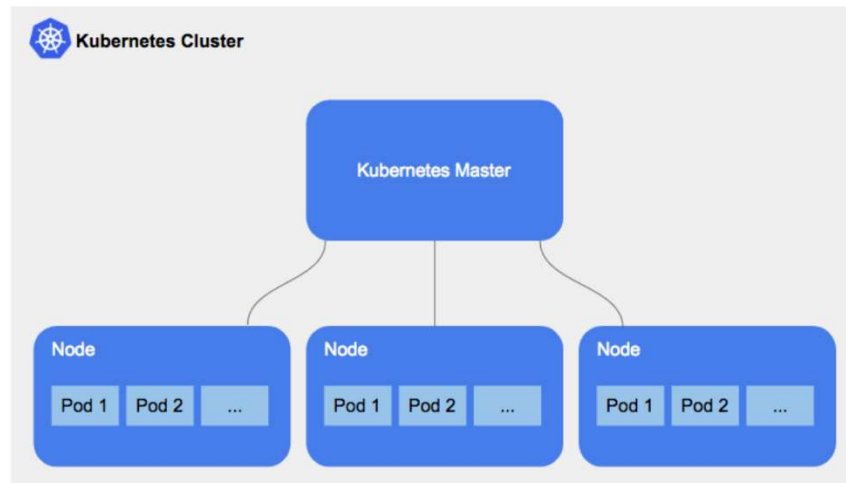
claims and obtains benefits from the control of the system as a whole, such that those components are put into use by Amazon and the steps are deemed to be performed by Amazon. Moreover, Amazon is the final assembler of the infringing system because it controls, assembles, and combines the components to form the infringing system.

117. The Accused Products include a computer network for providing an information delivery service for a plurality of participants, each participant having connections to at least three neighbor participants. For example, Amazon states that AWS “is the world’s most comprehensive and broadly adopted cloud platform, offering over 200 fully featured services from data centers globally. Millions of customers—including the fastest-growing startups, largest enterprises, and leading government agencies—are using AWS to lower costs, become more agile, and innovate faster.” Ex. 8 at 1, <https://aws.amazon.com/what-is-aws/>.

118. In the networks provided by the Accused Products, an originating participant sends data to the other participants by sending the data through each of its connections to its neighbor participants and each participant sends data that it receives from a neighbor participant to its other neighbor participants. For example, Amazon EKS groups machines using clusters, which are comprised on physical or virtual machines called nodes. Ex. 11, <https://medium.com/kubecost/understanding-kubernetes-cluster-autoscaling-675099a1db92>. The computers (e.g., physical or virtual machines) send messages between each other for requests for resources, for example. Furthermore, AWS utilizes mesh network architectures which involves the computers in the mesh network sending messages and relaying messages between each other as peers. Examples of these mesh networks are generated using VPC peering and App Mesh.

The Basics

Clusters are how Kubernetes groups machines. They are comprised of *Nodes* (individual machines, oftentimes virtual) which run *Pods*. *Pods* have containers that request resources such as CPU, Memory, and GPU. **The Cluster Autoscaler adds or removes *Nodes* in a *Cluster* based on resource requests from *Pods*.**



A high-level overview of a Kubernetes cluster

Id.

119. As discussed above, the Accused Products use networks that are m-regular, where m is the exact number of neighbor participants of each participant and the number of participants is at least two greater than m , thus resulting in a non-complete graph. The connections between the participants are peer-to-peer connections.

120. The Accused Products' network is formed through a broadcast channel that overlays an underlying network. For example, Accused Products use the `broadcastGroupMembershipUpdate()` function to form the network, which “[t]riggers delivery of an updated group membership list to player group. Specify which membership to broadcast (`groupIdToBroadcast`) and the group to receive the update (`targetGroupId`).” Ex. 26 at 225-26, <https://docs.aws.amazon.com/gamelift/latest/developerguide/gamelift-dg.pdf>.

121. The Accused Products' information delivery service is provided by at least one information delivery service application program executing on each computer of the computer network that interacts with the broadcast channel and participants can join and leave the network using the broadcast channel. Amazon operates information delivery services on its AWS platform, discussed above, such as Amazon Prime Video, Amazon Luna, Amazon Twitch, and CloudFront. For example, Twitch includes software apps, where users can join and leave the Twitch network using a broadcast channel used to communicate between Twitch servers and applications.

122. Amazon's infringement of the '966 Patent has injured and continues to injure Acceleration Bay in an amount to be proven at trial, but not less than a reasonable royalty. Amazon's actions are willful, blatant and in egregious disregard for Acceleration Bay's patent rights. Amazon's infringement has caused and is continuing to cause damage and irreparable injury to Acceleration Bay, and Acceleration Bay will continue to suffer damage and irreparable injury unless and until that infringement is enjoined by this Court.

123. Amazon acted recklessly, willfully, wantonly, and deliberately engaged in acts of infringement of the '966 Patent, justifying an award to Acceleration Bay of increased damages under 35 U.S.C. § 284, and attorney's fees and costs incurred under 35 U.S.C. § 285.

COUNT IV
(Indirect Infringement of the '966 Patent)

124. Acceleration Bay repeats, realleges, and incorporates by reference, as if fully set forth herein, the allegations of the preceding paragraphs.

125. As discussed above, Amazon knew about Acceleration Bay's patents, including the '966 Patent, and was informed it was infringing, by no later than March 13, 2019, and further knows about its infringement of the '966 Patent from its receipt of this Complaint.

126. In addition to directly infringing the '966 Patent, as discussed above with respect to Count III, Amazon knew or was willfully blind to the fact that it was inducing infringement of the '966 Patent under 35 U.S.C. § 271(b) by instructing, encouraging, directing, and requiring third parties, including its customers, to directly infringe by using and installing the Accused Products in the United States.

127. Amazon contributorily infringes the '966 Patent pursuant to 35 U.S.C. § 271(c) because it has provided software, that acts as a material component of the claimed inventions of the '966 Patent. In particular, Amazon knows that its products are particularly suited to be used in an infringing manner. Amazon has known or was willfully blind to the fact that it is contributing to the infringement of one or more claims of the '966 Patent. Amazon's software is not a staple article or commodity of commerce suitable for substantial noninfringing use.

128. Amazon knowingly and actively aided and abetted, encouraged, and contributed to the direct infringement of the '966 Patent by instructing and encouraging its customers, purchasers, users, developers, vendors, partners, and manufacturers to meet the elements of the '966 Patent with the Accused Products, as described above. Such instructions and encouragement included, but is not limited to, advising third parties to use the Accused Products in an infringing manner through direct communications, training and support materials, and customer support regarding how to install and configure the Accused Products, by advertising and promoting the use of the Accused Products in an infringing manner, and distributing release notes, webinars, guidelines, videos, manuals, white papers, and trainings to third parties on how the Accused Products must be used. *See, e.g.*, Ex. 10 (EKS User Guide) (including instructions to set up and configure EKS); Ex. 9 (Step-by-step guide to “[d]eploy an overlay network for regulated and sensitive workloads on AWS.”); Ex. 13 (AWS EC2 User Guide) (including

instructions to set up and configure EC2); Ex. 15 (AWS App Mesh User Guide) (including instructions to set up and configure App Mesh); Ex. 16 (AWS Lambda website); Ex. 19 (AWS Lambda Developer Guide) (including instructions to set up and configure Lambda); Ex. 20 (Amazon CloudFront Developer Guide) (including instructions to set up and configure CloudFront); Ex. 22 (VPC Peering guide) (including instructions to set up and configure VPC Peering); Ex. 23 (AWS Whitepaper on “Building a Scalable and Secure Multi-VPC AWS Network Infrastructure”); Ex. 24 (ELB guide) (including instructions to set up and configure Elastic Load Balancing); Ex. 25 (Lumberyard User Guide) (including instructions to set up and configure Lumberyard); Ex. 27 (AWS Website) (including instructions to set up and configure GridMate); Ex. 26 (Amazon GameLift Developer Guide) (including instructions to set up and configure GameLift); Ex. 33 (AWS General Reference) (including instructions to set up and configure AWS); Ex. 36 (EC2 Auto Scaling User Guide) (including instructions to set up and configure Auto Scaling).

129. Amazon’s indirect infringement of the ’966 Patent has injured and continues to injure Acceleration Bay in an amount to be proven at trial, but not less than a reasonable royalty. Amazon’s actions are willful, blatant, and in egregious disregard for Acceleration Bay’s patent rights. Amazon’s indirect infringement has caused and is continuing to cause damage and irreparable injury to Acceleration Bay, and Acceleration Bay will continue to suffer damage and irreparable injury unless and until that infringement is enjoined by this Court.

130. Amazon acted recklessly, willfully, wantonly, and deliberately engaged in acts of indirect infringement of the ’966 Patent, justifying an award to Acceleration Bay of increased damages under 35 U.S.C. § 284, and attorney’s fees and costs incurred under 35 U.S.C. § 285.

COUNT V
(Direct Infringement of the '634 Patent)

131. Acceleration Bay repeats, realleges, and incorporates by reference, as if fully set forth herein, the allegations of the preceding paragraphs, as set forth above.

132. Amazon has infringed and continues to infringe at least Claim 25 of the '634 Patent in violation of 35 U.S.C. § 271(a).

133. Amazon's infringement is based upon literal infringement or infringement under the doctrine of equivalents, or both.

134. Amazon's acts of making, using, importing, selling, and offering for sale the Accused Products has been without the permission, consent, authorization, or license of Acceleration Bay.

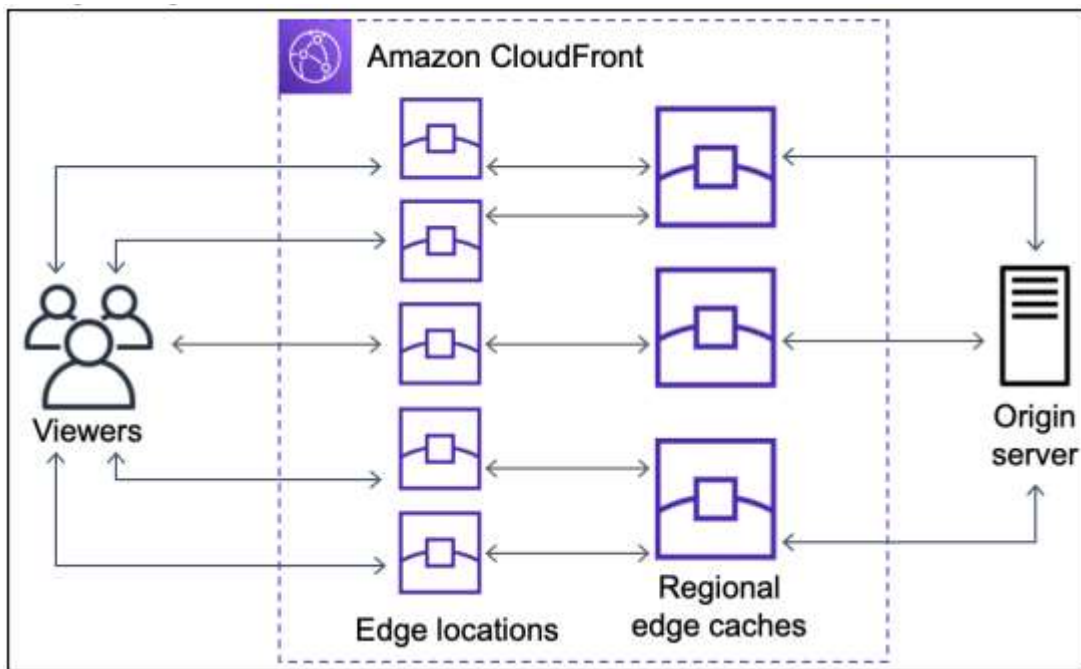
135. To the extent specific components are provided by or steps performed by Amazon's customers or partners, Amazon directs and controls the systems and methods of the claims and obtains benefits from the control of the system as a whole, such that those components are put into use by Amazon and the steps are deemed to be performed by Amazon. Moreover, Amazon is the final assembler of the infringing system because it controls, assembles, and combines the components to form the infringing system.

136. The Accused Products include a non-routing table based computer network having a plurality of participants, each participant being an application program, and each participant having connections to at least three neighbor participants. For example, Amazon CloudFront is a content delivery network that does not use routing tables. Amazon CloudFront "peers with thousands of Tier 1/2/3 telecom carriers globally, is well connected with all major access networks for optimal performance, and has hundreds of terabits of deployed capacity. CloudFront Edge locations are connected to the AWS Regions through the AWS network

backbone - fully redundant, multiple 100GbE parallel fiber that circles the globe and links with tens of thousands of networks for improved origin fetches and dynamic content acceleration.”

Ex. 21 at 1-2, <https://aws.amazon.com/cloudfront/features/?whats-new-cloudfront>. Amazon CloudFront includes a global network of “310+ Points of Presence (300+ Edge locations and 13 regional mid-tier caches) in 90+ cities across 47 countries.” *Id.* at 2.

137. In the networks of the Accused Products, an originating participant sends data to the other participants by sending the data through each of its connections to its neighbor participants and each participant sends data that it receives from a neighbor participant to its other neighbor participants. Shown below, Amazon CloudFront distributes data between origin servers, regional edge caches, and edge locations located all over the world.



Ex. 20 at 7, https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/AmazonCloudFront_DevGuide.pdf.

138. The Accused Products include process data which is numbered sequentially so that data received out of order can be queued and rearranged. For example, Amazon CloudFront

processes data such as multimedia content, which is segmented and “also generates manifest files, which describe in a specific order what segments to play and when.” *Id.* at 318. Media fragments received out of order may be queued and rearranged using a manifest file “that delivers the fragments in the right order can be combined, to reduce the load on [customers’] origin server.” *Id.* at 4.

139. As discussed above, the Accused Products use m-regular networks, where m is the exact number of neighbor participants of each participant and the number of participants is at least two greater than m thus resulting in a non-complete graph. The connections between the participants are peer-to-peer connections.

140. The Accused Products’ networks are dynamic networks which overlay an underlying network and participants can interact with a broadcast channel with a channel type and channel instance to join and leave the network. For example, application programs form overlay networks which operate at the application layer as an overly network on the AWS platform, as discussed above. For example, Amazon uses AWS to operate Amazon Prime Video, Amazon Luna, Amazon Twitch, and CloudFront. Twitch includes software apps, where users can join and leave the Twitch network using a broadcast channel used to communicate between Twitch servers and applications. Additionally, Amazon Prime Video allows users to choose video streaming content using the broadcast channel.

141. Amazon’s infringement of the ’634 Patent has injured and continues to injure Acceleration Bay in an amount to be proven at trial, but not less than a reasonable royalty.

142. Amazon’s infringement of the ’634 Patent has injured and continues to injure Acceleration Bay in an amount to be proven at trial, but not less than a reasonable royalty. Amazon’s actions are willful, blatant and in egregious disregard for Acceleration Bay’s patent

rights. Amazon's infringement has caused and is continuing to cause damage and irreparable injury to Acceleration Bay, and Acceleration Bay will continue to suffer damage and irreparable injury unless and until that infringement is enjoined by this Court.

143. Amazon acted recklessly, willfully, wantonly, and deliberately engaged in acts of infringement of the '634 Patent, justifying an award to Acceleration Bay of increased damages under 35 U.S.C. § 284, and attorney's fees and costs incurred under 35 U.S.C. § 285.

COUNT VI
(Indirect Infringement of the '634 Patent)

144. Acceleration Bay repeats, realleges, and incorporates by reference, as if fully set forth herein, the allegations of the preceding paragraphs.

145. As discussed above, Amazon knew about Acceleration Bay's patents, including the '634 Patent, and was informed it was infringing, by no later than March 13, 2019, and further knows about its infringement of the '634 Patent from its receipt of this Complaint.

146. In addition to directly infringing the '634 Patent, as discussed above with respect to Count V, Amazon knew or was willfully blind to the fact that it was inducing infringement of the '634 Patent under 35 U.S.C. § 271(b) by instructing, encouraging, directing, and requiring third parties, including its customers, to directly infringe by using and installing the Accused Products in the United States.

147. Amazon contributorily infringes the '634 Patent pursuant to 35 U.S.C. § 271(c) because it has provided software, that acts as a material component of the claimed inventions of the '634 Patent. In particular, Amazon knows that its products are particularly suited to be used in an infringing manner. Amazon has known or was willfully blind to the fact that it is contributing to the infringement of one or more claims of the '634 Patent. Amazon's software is not a staple article or commodity of commerce suitable for substantial noninfringing use.

148. Amazon knowingly and actively aided and abetted, encouraged, and contributed to the direct infringement of the '634 Patent by instructing and encouraging its customers, purchasers, users, developers, vendors, partners, and manufacturers to meet the elements of the '634 Patent with the Accused Products, as described above. Such instructions and encouragement included, but is not limited to, advising third parties to use the Accused Products in an infringing manner through direct communications, training and support materials, and customer support regarding how to install and configure the Accused Products, by advertising and promoting the use of the Accused Products in an infringing manner, and distributing release notes, webinars, guidelines, videos, manuals, white papers, and trainings to third parties on how the Accused Products must be used. *See, e.g.*, Ex. 10 (EKS User Guide) (including instructions to set up and configure EKS); Cohesive Networks VNS3 on AWS - Quick Start.pdf (Step-by-step guide to “[d]eploy an overlay network for regulated and sensitive workloads on AWS.”); ecs-ug.pdf (AWS EC2 User Guide) (including instructions to set up and configure EC2); Ex. 15 (AWS App Mesh User Guide) (including instructions to set up and configure App Mesh); Serverless Computing - AWS Lambda - Amazon Web Services.pdf (AWS Lambda website); Ex. 19 (AWS Lambda Developer Guide) (including instructions to set up and configure Lambda); Ex. 20 (Amazon CloudFront Developer Guide) (including instructions to set up and configure CloudFront); Ex. 22 (VPC Peering guide) (including instructions to set up and configure VPC Peering); building-scalable-secure-multi-vpc-network-infrastructure.pdf (AWS Whitepaper on “Building a Scalable and Secure Multi-VPC AWS Network Infrastructure”); Ex. 24 (ELB guide) (including instructions to set up and configure Elastic Load Balancing); Ex. 25 (Lumberyard User Guide) (including instructions to set up and configure Lumberyard); Building Battle-Tested Network Transport _ AWS Game Tech Blog.pdf (AWS Website) (including instructions to set

up and configure GridMate); Ex. 26 (Amazon GameLift Developer Guide) (including instructions to set up and configure GameLift); Ex. 33 (AWS General Reference) (including instructions to set up and configure AWS); Ex. 36 (EC2 Auto Scaling User Guide) (including instructions to set up and configure Auto Scaling).

149. Amazon's indirect infringement of the '634 Patent has injured and continues to injure Acceleration Bay in an amount to be proven at trial, but not less than a reasonable royalty. Amazon's actions are willful, blatant, and in egregious disregard for Acceleration Bay's patent rights. Amazon's indirect infringement has caused and is continuing to cause damage and irreparable injury to Acceleration Bay, and Acceleration Bay will continue to suffer damage and irreparable injury unless and until that infringement is enjoined by this Court.

150. Amazon acted recklessly, willfully, wantonly, and deliberately engaged in acts of indirect infringement of the '634 Patent, justifying an award to Acceleration Bay of increased damages under 35 U.S.C. § 284, and attorney's fees and costs incurred under 35 U.S.C. § 285.

COUNT VII
(Direct Infringement of the '069 Patent)

151. Acceleration Bay repeats, realleges, and incorporates by reference, as if fully set forth herein, the allegations of the preceding paragraphs, as set forth above.

152. Amazon has infringed and continues to infringe at least Claim 1 of the '069 Patent in violation of 35 U.S.C. § 271(a).

153. Amazon's infringement is based upon literal infringement or infringement under the doctrine of equivalents, or both.

154. Amazon's acts of making, using, importing, selling, and offering for sale the Accused Products has been without the permission, consent, authorization, or license of Acceleration Bay.

155. To the extent specific components are provided by or steps performed by Amazon's customers or partners, Amazon directs and controls the systems and methods of the claims and obtains benefits from the control of the system as a whole, such that those components are put into use by Amazon and the steps are deemed to be performed by Amazon. Moreover, Amazon is the final assembler of the infringing system because it controls, assembles, and combines the components to form the infringing system.

156. The Accused Products include a computer-based, non-switch based method for adding nodes to a graph that is m -regular and m -connected to maintain the graph as m -regular, where m is four or greater. As discussed above, the Accused Products use m -regular, incomplete networks. For example, Amazon uses AWS to operate its broadcast services such as Amazon Prime Video, Amazon Luna, Amazon Twitch, and CloudFront.

157. The Accused Products identify p pairs of nodes of the graph that are connected, where p is one half of m , wherein a seeking node contacts a fully connected portal node, which in turn sends an edge connection request to a number of randomly selected neighboring nodes to which the seeking node is to connect.

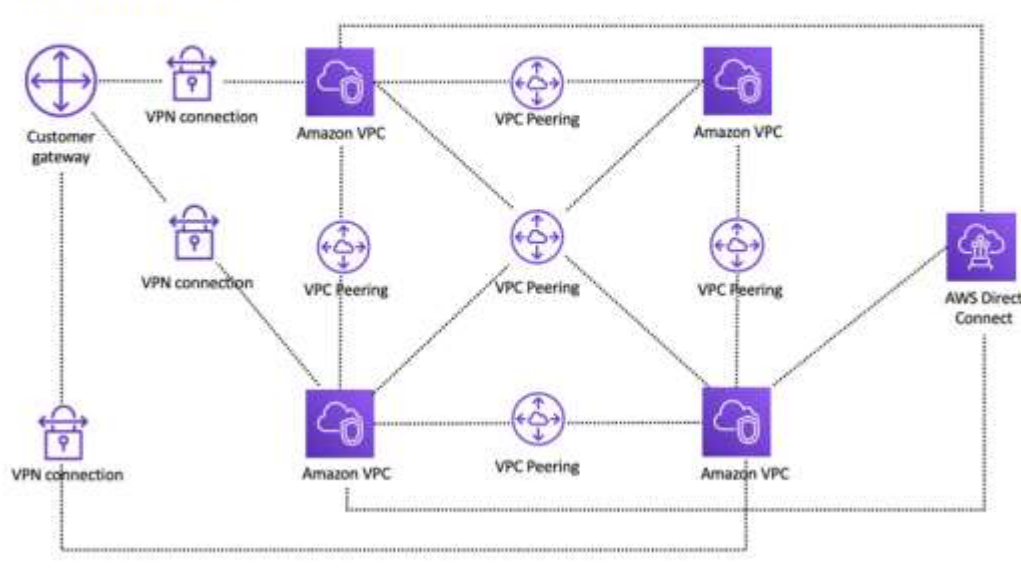
158. For example, the Accused Products utilize VPC peering, which results in "point-to-point connectivity" of VPCs using "full bidirectional connectivity between the VPCs. This peering connection is used to route traffic between the VPCs. VPCs across accounts and AWS Regions can also be peered together." Ex. 23 at 4,

<https://docs.aws.amazon.com/whitepapers/latest/building-scalable-secure-multi-vpc-network-infrastructure/building-scalable-secure-multi-vpc-network-infrastructure.pdf#welcome>.

159. Interconnecting multiple VPCs (e.g., tens to hundreds) at scale "results in a mesh of hundreds to thousands of peering connections." For example, a 100 VPCs in a full mesh

peering configuration would require “4,950 peering connections $[n(n-1)/2]$ where n =total number of VPCs.” *Id.* However, the VPCs are limited to a maximum of 125 connections, making the network incomplete.

At scale, when you have tens to hundreds of VPCs, interconnecting them with peering results in a mesh of hundreds to thousands of peering connections, which are difficult to manage and scale. For example, if you have 100 VPCs and you want to setup a full mesh peering between them, it will take 4,950 peering connections $[n(n-1)/2]$ where n =total number of VPCs. There is a **maximum limit** of up to 125 active peering connections per VPC.

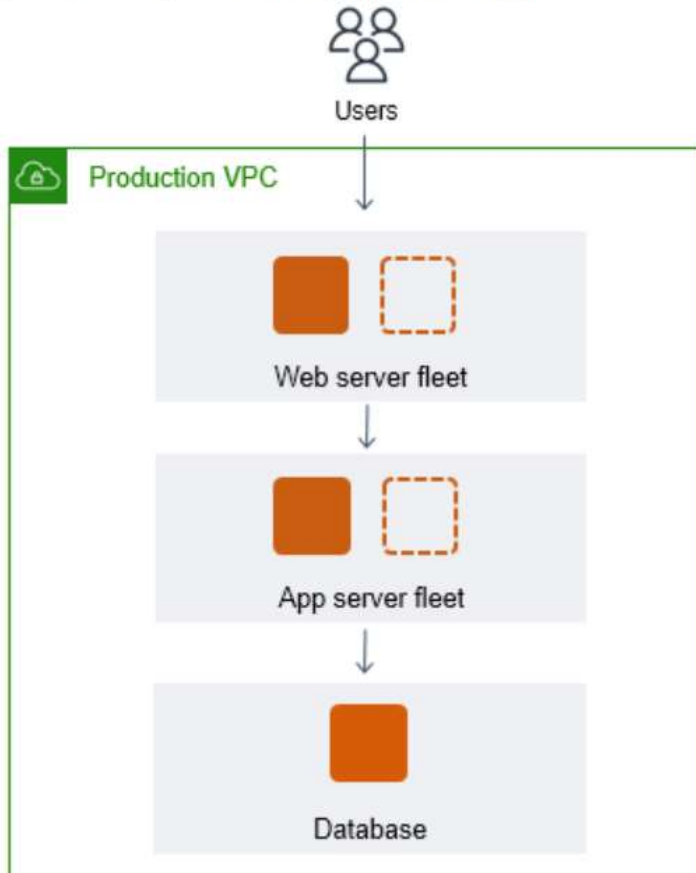


Id.

160. The Accused Products disconnect the nodes of each identified pair from each other and connect each node of the identified pairs of nodes to the seeking node. For example, Amazon VPC Peering supports Auto Scaling, which supports adding “new instances to the application only when necessary, and terminate them when they’re no longer needed.” Ex. 36 at 3-5, <https://docs.aws.amazon.com/autoscaling/ec2/userguide/as-dg.pdf#asg-in-vpc>. VPC Peering Auto Scaling “works essentially the same way as it does on Amazon EC2 and supports the same set of features,” permitting the m-regular, incomplete network to be maintained as participants are added and removed. *Id.* at 94-96.

Example: Web app architecture

In a common web app scenario, you run multiple copies of your app simultaneously to cover the volume of your customer traffic. These multiple copies of your application are hosted on identical EC2 instances (cloud servers), each handling customer requests.



Amazon EC2 Auto Scaling manages the launch and termination of these EC2 instances on your behalf. You define a set of criteria (such as an Amazon CloudWatch alarm) that determines when the Auto Scaling group launches or terminates EC2 instances. Adding Auto Scaling groups to your network architecture helps make your application more highly available and fault tolerant.

You can create as many Auto Scaling groups as you need. For example, you can create an Auto Scaling group for each tier.

Id. at 3-5.

161. Amazon’s infringement of the ’069 Patent has injured and continues to injure Acceleration Bay in an amount to be proven at trial, but not less than a reasonable royalty. Amazon’s actions are willful, blatant and in egregious disregard for Acceleration Bay’s patent rights. Amazon’s infringement has caused and is continuing to cause damage and irreparable

injury to Acceleration Bay, and Acceleration Bay will continue to suffer damage and irreparable injury unless and until that infringement is enjoined by this Court.

162. Amazon acted recklessly, willfully, wantonly, and deliberately engaged in acts of infringement of the '069 Patent, justifying an award to Acceleration Bay of increased damages under 35 U.S.C. § 284, and attorney's fees and costs incurred under 35 U.S.C. § 285.

COUNT VIII
(Indirect Infringement of the '069 Patent)

163. Acceleration Bay repeats, realleges, and incorporates by reference, as if fully set forth herein, the allegations of the preceding paragraphs.

164. As discussed above, Amazon knew about Acceleration Bay's patents, including the '069 Patent, and was informed it was infringing, by no later than March 13, 2019, and further knows about its infringement of the '069 Patent from its receipt of this Complaint.

165. In addition to directly infringing the '069 Patent, as discussed above with respect to Count VII, Amazon knew or was willfully blind to the fact that it was inducing infringement of the '069 Patent under 35 U.S.C. § 271(b) by instructing, encouraging, directing, and requiring third parties, including its customers, to directly infringe by using and installing the Accused Products in the United States.

166. Amazon contributorily infringes the '069 Patent pursuant to 35 U.S.C. § 271(c) because it has provided software, that acts as a material component of the claimed inventions of the '069 Patent. In particular, Amazon knows that its products are particularly suited to be used in an infringing manner. Amazon has known or was willfully blind to the fact that it is contributing to the infringement of one or more claims of the '069 Patent. Amazon's software is not a staple article or commodity of commerce suitable for substantial noninfringing use.

167. Amazon knowingly and actively aided and abetted, encouraged, and contributed to the direct infringement of the '069 Patent by instructing and encouraging its customers, purchasers, users, developers, vendors, partners, and manufacturers to meet the elements of the '069 Patent with the Accused Products, as described above. Such instructions and encouragement included, but is not limited to, advising third parties to use the Accused Products in an infringing manner through direct communications, training and support materials, and customer support regarding how to install and configure the Accused Products, by advertising and promoting the use of the Accused Products in an infringing manner, and distributing release notes, webinars, guidelines, videos, manuals, white papers, and trainings to third parties on how the Accused Products must be used. *See, e.g.*, Ex. 10 (EKS User Guide) (including instructions to set up and configure EKS); Ex. 9 (Step-by-step guide to “[d]eploy an overlay network for regulated and sensitive workloads on AWS.”); Ex. 13 (AWS EC2 User Guide) (including instructions to set up and configure EC2); Ex. 15 (AWS App Mesh User Guide) (including instructions to set up and configure App Mesh); Ex. 16 (AWS Lambda website); Ex. 19 (AWS Lambda Developer Guide) (including instructions to set up and configure Lambda); Ex. 20 (Amazon CloudFront Developer Guide) (including instructions to set up and configure CloudFront); Ex. 22 (VPC Peering guide) (including instructions to set up and configure VPC Peering); Ex. 23 (AWS Whitepaper on “Building a Scalable and Secure Multi-VPC AWS Network Infrastructure”); Ex. 24 (ELB guide) (including instructions to set up and configure Elastic Load Balancing); Ex. 25 (Lumberyard User Guide) (including instructions to set up and configure Lumberyard); Ex. 27 (AWS Website) (including instructions to set up and configure GridMate); Ex. 26 (Amazon GameLift Developer Guide) (including instructions to set up and configure GameLift); Ex. 33 (AWS General Reference) (including instructions to set up and

configure AWS); Ex. 36 (EC2 Auto Scaling User Guide) (including instructions to set up and configure Auto Scaling).

168. Amazon's indirect infringement of the '069 Patent has injured and continues to injure Acceleration Bay in an amount to be proven at trial, but not less than a reasonable royalty. Amazon's actions are willful, blatant, and in egregious disregard for Acceleration Bay's patent rights. Amazon's indirect infringement has caused and is continuing to cause damage and irreparable injury to Acceleration Bay, and Acceleration Bay will continue to suffer damage and irreparable injury unless and until that infringement is enjoined by this Court.

169. Amazon acted recklessly, willfully, wantonly, and deliberately engaged in acts of indirect infringement of the '069 Patent, justifying an award to Acceleration Bay of increased damages under 35 U.S.C. § 284, and attorney's fees and costs incurred under 35 U.S.C. § 285.

COUNT IX
(Direct Infringement of the '147 Patent)

170. Acceleration Bay repeats, realleges, and incorporates by reference, as if fully set forth herein, the allegations of the preceding paragraphs, as set forth above.

171. Amazon has infringed and continues to infringe at least Claim 6 of the '147 Patent in violation of 35 U.S.C. § 271(a).

172. Amazon's infringement is based upon literal infringement or infringement under the doctrine of equivalents, or both.

173. Amazon's acts of making, using, importing, selling, and offering for sale the Accused Products has been without the permission, consent, authorization, or license of Acceleration Bay.

174. To the extent specific components are provided by or steps performed by Amazon's customers or partners, Amazon directs and controls the systems and methods of the

claims and obtains benefits from the control of the system as a whole, such that those components are put into use by Amazon and the steps are deemed to be performed by Amazon. Moreover, Amazon is the final assembler of the infringing system because it controls, assembles, and combines the components to form the infringing system.

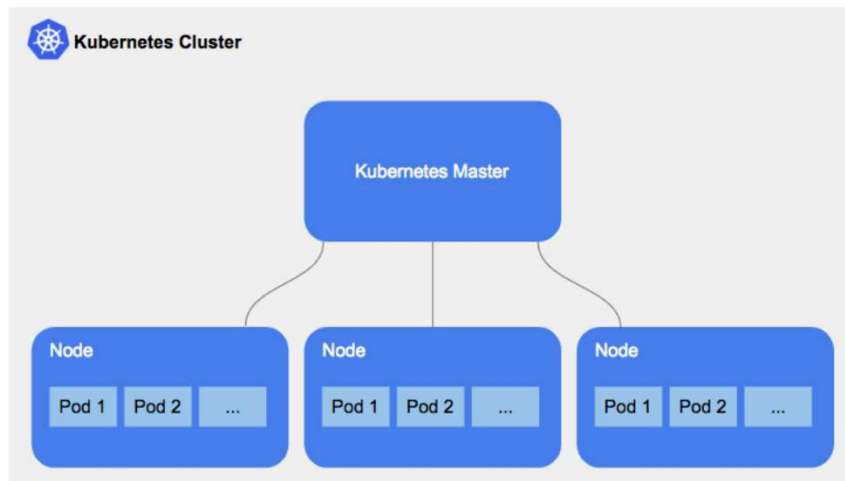
175. The Accused Products practice a method for healing a disconnection of a first computer from a second computer, the computers being connected to a broadcast channel, said broadcast channel being an m -regular graph where m is at least 3. AWS is a scalable cloud infrastructure. AWS includes EKS, which is “a managed service that [customers] can use to run Kubernetes on AWS without needing to install, operate, and maintain [their] own Kubernetes control plane or nodes. Kubernetes is an open-source system for automating the deployment, scaling, and management of containerized applications.” Ex. 10 at 1, <https://docs.aws.amazon.com/eks/latest/userguide/eks-ug.pdf>. Amazon EKS “[r]uns and scales the Kubernetes control plane across multiple AWS Availability Zones to ensure high availability,” “[a]utomatically scales control plane instances based on load, detects and replaces unhealthy control plane instances,” “[i]s integrated with many AWS services to provide scalability and security,” including Amazon ECR for container images, Elastic Load Balancing for load distribution, IAM for authentication, and Amazon VPC for isolation.” *Id.*

176. This functionality permits the m -regular, incomplete network to be maintained as participants are added. Amazon EKS “[a]ctively monitors the load on control plane instances and automatically scales them to ensure high performance,” “[a]utomatically detects and replaces unhealthy control plane instances, restarting them across the Availability Zones within the AWS Region as needed,” and “[l]everages the architecture of AWS Regions in order to maintain high availability.” *Id.*

177. The Accused Products attempt to send a message from the first computer to the second computer. For example, Amazon EKS groups machines using clusters, which are comprised on physical or virtual machines called nodes. Ex. 11, <https://medium.com/kubecost/understanding-kubernetes-cluster-autoscaling-675099a1db92>. The computers (e.g., physical or virtual machines) send messages between each other for requests for resources, for example. Furthermore, AWS utilizes mesh network architectures which involves the computers in the mesh network sending messages between each other. Examples of these mesh networks are generated using VPC peering and App Mesh.

The Basics

Clusters are how Kubernetes groups machines. They are comprised of *Nodes* (individual machines, oftentimes virtual) which run *Pods*. *Pods* have containers that request resources such as CPU, Memory, and GPU. **The Cluster Autoscaler adds or removes *Nodes* in a *Cluster* based on resource requests from *Pods*.**



A high-level overview of a Kubernetes cluster

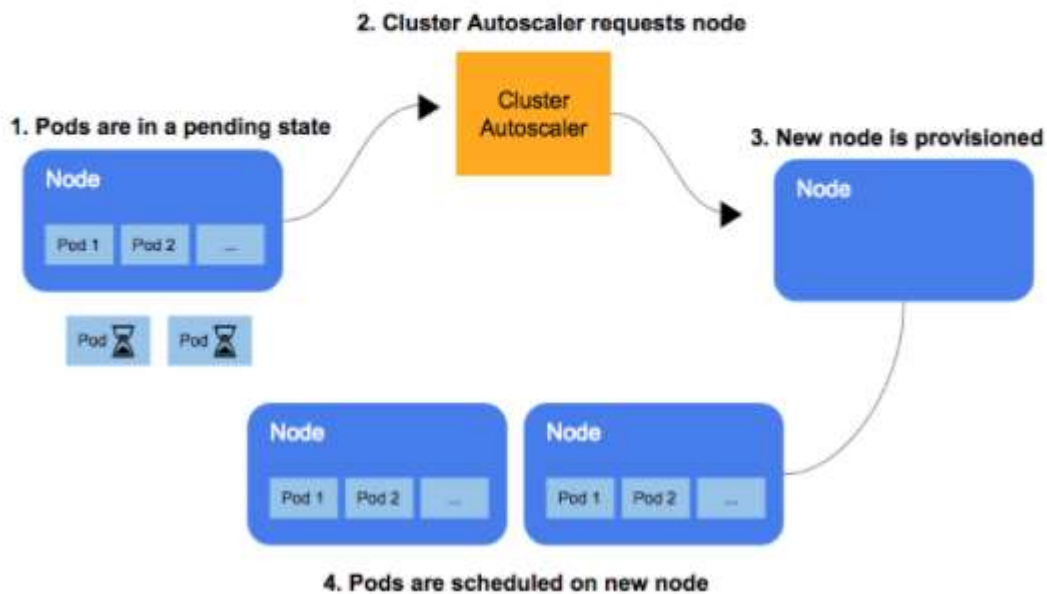
Ex. 11, <https://medium.com/kubecost/understanding-kubernetes-cluster-autoscaling-675099a1db92>.

178. When an attempt to broadcast a message is unsuccessful, the Accused Products broadcast from a first computer a connection port search message indicating that the first

computer needs a connection. The Accused Products use autoscaling to add and remove computing resources based on the needs of the applications. For example, the Accused Products include a Cluster Autoscaler, which scales the cluster by adding new nodes when a connection is needed, such as a request for resources. *Id.* The Accused Products use a cluster autoscaler algorithm to check for pending pods and will provision a new node as necessary. *Id.*

When Does The Cluster Autoscaler Add Capacity?

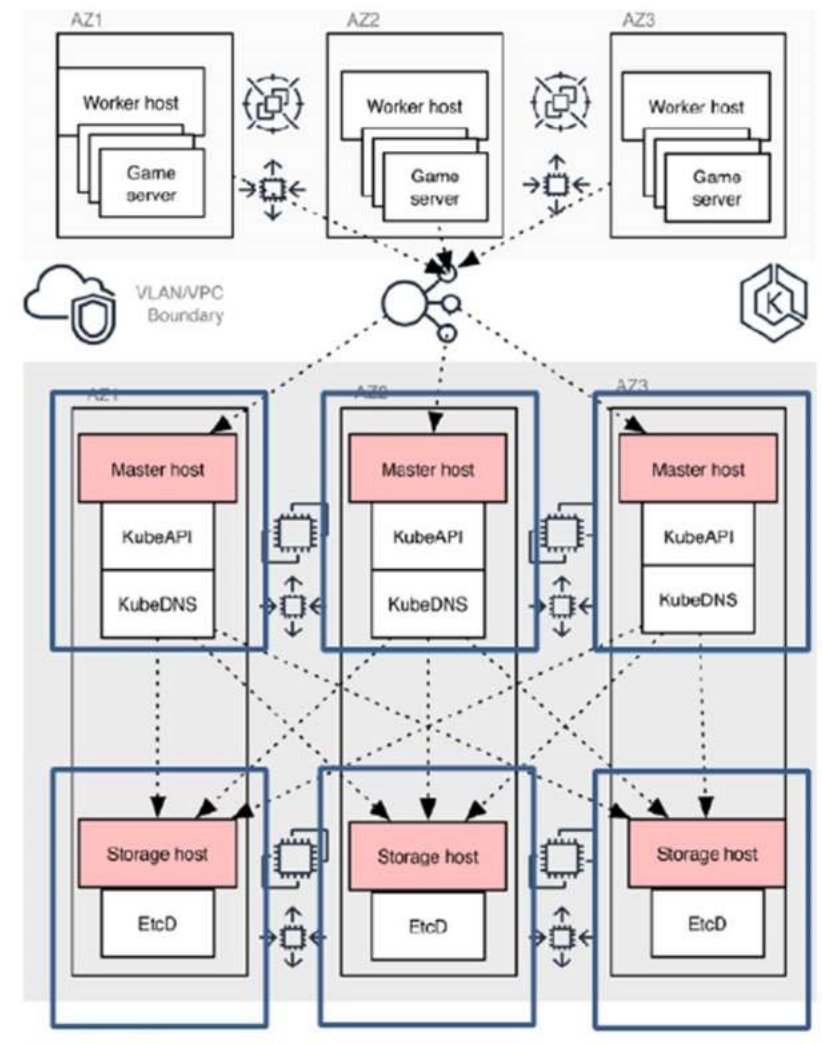
The cluster autoscaler increases the size of the cluster when there are pods that are not able to be scheduled due to resource shortages. It can be configured to not scale up or down past a certain number of machines. Here is an overview of how it makes scaling decisions:



Ex. 11, <https://medium.com/kubecost/understanding-kubernetes-cluster-autoscaling-675099a1db92>.

179. The Accused Products have a third computer not already connected to said first computer respond to said connection port search message in a manner as to maintain an m-

regular graph. For example, the Accused Products request and add a newly provisioned node to the cluster in response to the connection port search message to maintain an m-regular graph. *Id.* Shown below, the Accused Products operate on availability zones in various regions and interconnect to maintain m-regularity.



Ex. 37, <https://aws.amazon.com/blogs/compute/running-your-game-servers-at-scale-for-up-to-90-lower-compute-cost/>.

180. As discussed above, the Accused Products use m-regular, incomplete networks. For example, Amazon uses AWS to operate its broadcast services such as Amazon Prime Video, Amazon Luna, Amazon Twitch, and CloudFront.

181. Amazon's infringement of the '147 Patent has injured and continues to injure Acceleration Bay in an amount to be proven at trial, but not less than a reasonable royalty. Amazon's actions are willful, blatant and in egregious disregard for Acceleration Bay's patent rights. Amazon's infringement has caused and is continuing to cause damage and irreparable injury to Acceleration Bay, and Acceleration Bay will continue to suffer damage and irreparable injury unless and until that infringement is enjoined by this Court.

182. Amazon acted recklessly, willfully, wantonly, and deliberately engaged in acts of infringement of the '147 Patent, justifying an award to Acceleration Bay of increased damages under 35 U.S.C. § 284, and attorney's fees and costs incurred under 35 U.S.C. § 285.

COUNT X
(Indirect Infringement of the '147 Patent)

183. Acceleration Bay repeats, realleges, and incorporates by reference, as if fully set forth herein, the allegations of the preceding paragraphs.

184. As discussed above, Amazon knew about Acceleration Bay's patents, including the '147 Patent, and was informed it was infringing, by no later than March 13, 2019, and further knows about its infringement of the '147 Patent from its receipt of this Complaint.

185. In addition to directly infringing the '147 Patent, as discussed above with respect to Count IX, Amazon knew or was willfully blind to the fact that it was inducing infringement of the '147 Patent under 35 U.S.C. § 271(b) by instructing, encouraging, directing, and requiring third parties, including its customers, to directly infringe by using and installing the Accused Products in the United States.

186. Amazon contributorily infringes the '147 Patent pursuant to 35 U.S.C. § 271(c) because it has provided software, that acts as a material component of the claimed inventions of the '147 Patent. In particular, Amazon knows that its products are particularly suited to be used

in an infringing manner. Amazon has known or was willfully blind to the fact that it is contributing to the infringement of one or more claims of the '147 Patent. Amazon's software is not a staple article or commodity of commerce suitable for substantial noninfringing use.

187. Amazon knowingly and actively aided and abetted, encouraged, and contributed to the direct infringement of the '147 Patent by instructing and encouraging its customers, purchasers, users, developers, vendors, partners, and manufacturers to meet the elements of the '147 Patent with the Accused Products, as described above. Such instructions and encouragement included, but is not limited to, advising third parties to use the Accused Products in an infringing manner through direct communications, training and support materials, and customer support regarding how to install and configure the Accused Products, by advertising and promoting the use of the Accused Products in an infringing manner, and distributing release notes, webinars, guidelines, videos, manuals, white papers, and trainings to third parties on how the Accused Products must be used. *See, e.g.*, Ex. 10 (EKS User Guide) (including instructions to set up and configure EKS); Ex. 9 (Step-by-step guide to “[d]eploy an overlay network for regulated and sensitive workloads on AWS.”); Ex. 13 (AWS EC2 User Guide) (including instructions to set up and configure EC2); Ex. 15 (AWS App Mesh User Guide) (including instructions to set up and configure App Mesh); Ex. 16 (AWS Lambda website); Ex. 19 (AWS Lambda Developer Guide) (including instructions to set up and configure Lambda); Ex. 20 (Amazon CloudFront Developer Guide) (including instructions to set up and configure CloudFront); Ex. 22 (VPC Peering guide) (including instructions to set up and configure VPC Peering); Ex. 23 (AWS Whitepaper on “Building a Scalable and Secure Multi-VPC AWS Network Infrastructure”); Ex. 24 (ELB guide) (including instructions to set up and configure Elastic Load Balancing); Ex. 25 (Lumberyard User Guide) (including instructions to set up and

configure Lumberyard); Ex. 27 (AWS Website) (including instructions to set up and configure GridMate); Ex. 26 (Amazon GameLift Developer Guide) (including instructions to set up and configure GameLift); Ex. 33 (AWS General Reference) (including instructions to set up and configure AWS); Ex. 36 (EC2 Auto Scaling User Guide) (including instructions to set up and configure Auto Scaling).

188. Amazon's indirect infringement of the '147 Patent has injured and continues to injure Acceleration Bay in an amount to be proven at trial, but not less than a reasonable royalty. Amazon's actions are willful, blatant, and in egregious disregard for Acceleration Bay's patent rights. Amazon's indirect infringement has caused and is continuing to cause damage and irreparable injury to Acceleration Bay, and Acceleration Bay will continue to suffer damage and irreparable injury unless and until that infringement is enjoined by this Court.

189. Amazon acted recklessly, willfully, wantonly, and deliberately engaged in acts of indirect infringement of the '147 Patent, justifying an award to Acceleration Bay of increased damages under 35 U.S.C. § 284, and attorney's fees and costs incurred under 35 U.S.C. § 285.

PRAYER FOR RELIEF

WHEREFORE, Acceleration Bay prays for judgment and relief as follows:

A. an entry of judgment holding that Amazon has infringed and is infringing the '344, '966, '147, '634, and '069 Patents; and has induced infringement and is inducing infringement of the '344, '966, '147, '634, and '069 Patents; and has contributorily infringed and continues to contribute to infringement of the '344, '966, '147, '634, and '069 Patents;

B. a preliminary and permanent injunction against Amazon and its officers, employees, agents, servants, attorneys, instrumentalities, and/or those in privity with them, from

infringing, inducing, and contributing to the infringement of the '344, '966, '147, '634, and '069 Patents and for all further and proper injunctive relief pursuant to 35 U.S.C. § 283;

C. an award to Acceleration Bay of such damages as it shall prove at trial against Amazon that is adequate to fully compensate Acceleration Bay for Amazon's infringement of the '344, '966, '147, '634, and '069 Patents; said damages to be no less than a reasonable royalty;

D. a determination that Amazon's infringement has been willful, wanton, and deliberate, and that the damages against it be trebled on this basis or for any other basis in accordance with the law;

E. an award to Acceleration Bay of increased damages under 35 U.S.C. § 284, including that Amazon willfully infringed the '344, '966, '147, '634, and '069 Patents;

F. a finding that this case is "exceptional" and an award to Acceleration Bay of its costs and reasonable attorneys' fees, as provided by 35 U.S.C. § 285;

G. an accounting of all infringing sales and revenues, together with post judgment interest and prejudgment interest from the first date of infringement of the '344, '966, '147, '634, and '069 Patents; and

H. such further and other relief as the Court may deem proper and just.

DEMAND FOR JURY TRIAL

Acceleration Bay demands a jury trial on all issues so triable.

POTTER ANDERSON & CORROON LLP

OF COUNSEL:

Paul J. Andre
Lisa Kobialka
James Hannah
Michael Lee
Christina Finn
KRAMER LEVIN NAFTALIS
& FRANKEL LLP
990 Marsh road
Menlo Park, CA 94025
(650) 752-1700

Aaron M. Frankel
Marcus Colucci
KRAMER LEVIN NAFTALIS
& FRANKEL LLP
1177 Avenue of the Americas
New York, NY 10036
(212) 715-9100

Date: July 6, 2022
10217462

By: /s/ Philip A. Rovner
Philip A. Rover (#3215)
Jonathan A. Choa (#5319)
Hercules Plaza
P.O. Box 951
Wilmington, Delaware 19801
(302) 984-6000
provner@potteranderson.com
jchoa@potteranderson.com

*Attorneys for Plaintiff
Acceleration Bay, LLC*