BEUS GILBERT McGRODER PLLC
ATTORNEYS AT LAW
701 NORTH 44TH STREET
PHOENIX, ARIZONA 85008-6504
TELEPHONE (480) 429-3000

Leo R. Beus / 002687 / lbeus@beusgilbert.com
Michael K. Kelly / 014203 / mkelly@beusgilbert.com
Christine N. Jones / 018425/ cjones@beusgilbert.com
Daniel J. Anderson / 030338 / danderson@beusgilbert.com

Attorneys for Plaintiff

UNITED STATES DISTRICT COURT

DISTRICT OF ARIZONA

Swarm Technology, LLC, an Arizona limited liability company, Plaintiff, v. Amazon.com, Inc., and Amazon Web Services, Inc., Defendants.	Case No.: CV-21-00438-PHX-DWL **PLAINTIFF'S FIRST AMENDED COMPLAINT FOR PATENT INFRINGEMENT** **Jury Trial Demanded**

1. Plaintiff Swarm Technology, an Arizona limited liability company ("Swarm"), brings this patent infringement action under Title 35 of the United States Code against Defendants Amazon.com, Inc. ("Amazon") and Amazon Web Services, Inc. ("AWS") (collectively, the "Amazon Defendants"). Plaintiff alleges the following upon personal knowledge and belief as to its own actions, and upon information and belief as to all other matters:

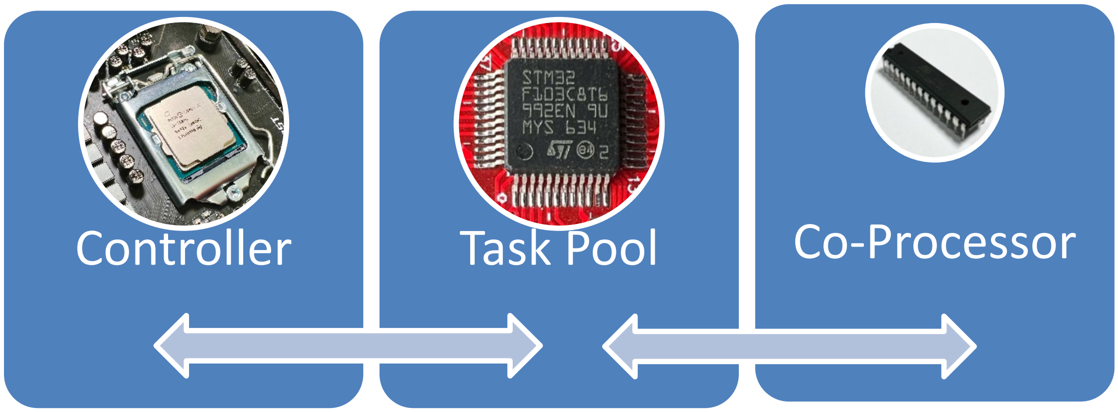
INTRODUCTION

2. Alfonso Íñiguez is the sole inventor of three (3) United States Patents, a Japanese patent, and has a number of additional patent applications currently pending in the United States, the European Union, India, China, and Hong Kong relating to a revolutionary new computer processing architecture.

3. Prior to Mr. Íñiguez' invention, conventional multiprocessors included a central processing unit ("CPU") and one or more co-processors (see below). That is, the processing "architecture" consisted of a primary controller which distributed tasks directly to a plurality of co-processors. This conventional approach is disadvantageous in that a significant amount of the CPU's processing cycles (bandwidth) is consumed by task distribution, while the co-processors often remain idle while waiting for a new task from the CPU.

Conventional Architecture

4. Mr. Íñiguez modified the arrangement of components (the architecture) of multiprocessing systems (see below) by interposing an intermediate device – the task pool – between the CPU and the co-processors. In addition, Mr. Íñiguez modified the way each of those components operate, both individually and in combination with each other. As a result, Mr. Íñiguez proposed a new multiprocessor system architecture which had never existed before.



Swarm Architecture

5. The United States Patent and Trademark Office has awarded, *inter alia*, the following Patents to Mr. Íñiguez: i) U.S. Patent No. 9,852,004 issued December 26, 2017, and entitled "System and Method for Parallel Processing Using Dynamically Configurable Proactive Co-Processing Cells" (the '004 Patent); ii) U.S. Patent No. 10,592,275 issued March 17, 2020, and entitled "System and Method for Swarm Collaborative Intelligence Using Dynamically Configurable Proactive Autonomous Agents" (the '275 Patent); and iii) U.S. Patent No. 9,146,777 issued September 29, 2015, and entitled "Parallel Processing With Solidarity Cells By Proactively Retrieving From a Task Pool a Matching Task for the Solidarity Cell to Process" (the '777 Patent). True and correct copies of the '004 Patent, the '275 Patent, and the '777 Patent (collectively, the "Patents-in-Suit") are attached as Exhibits A, B, and A2, respectively, and incorporated herein by this reference. As set forth more fully below, and in the claim charts attached as Exhibits W, X and Z, the Amazon Defendants infringe all claims of each of the Patents-in-Suit either directly, contributorily, or through inducement (35 U.S.C. § 271).

6. Alfonso and Alejandra Íñiguez founded Swarm Technology, LLC as an Arizona Limited Liability Company on January 17, 2014. Pursuant to written assignments from Mr. Íñiguez, the Patents-in-Suit are now owned by Swarm Technology, LLC.

7. In recent years the cloud computing industry, led by AWS, has migrated away from the traditional "primary/secondary" model – in which a central controller

directly controls a plurality of microprocessors – to a distributed "co-processing" model. The new co-processing model does not require direct communication between the controller and the co-processors. Instead, coordination between the controller (typically a desktop, laptop, or hand-held computer) and the co-processors involves an intermediary data structure referred to as a "task pool." A credentialed administrator uses the controller to populate the task pool with discrete tasks to be performed by the co-processors. Each co-processor proactively retrieves tasks directly from the task pool and notifies the task pool when each task is completed. This allows the controller to indirectly accomplish multiple tasks without having to expend unnecessary processing cycles directly supervising the co-processors.

8. The systems and methods used in Amazon's cloud computing products and services are precisely the same as those claimed in the Patents-in-Suit. Consequently, the Amazon Defendants are jointly and severally liable to Swarm for infringing the Patents-in-Suit.

THE PARTIES

9. Swarm Technology, LLC, is an Arizona limited liability company (Arizona Entity ID L18990310) with its principal place of business at 732 East Lehi Road, Mesa, Arizona 85203.

10. Alfonso Íñiguez is the inventor of the Patents-in-Suit, a Member of Swarm Technology, LLC, and a resident of Mesa, Arizona.

11. Alejandra Íñiguez is a Member of Swarm Technology, LLC, and a resident of Mesa, Arizona.

12. Alfonso and Alejandra Íñiguez are husband and wife and are the sole owners of Swarm Technology, LLC.

13. Upon information and belief, Defendant Amazon.com, Inc. is a Delaware corporation with its headquarters located in the State of Washington at 410 Terry Ave. North, Seattle, WA 98109-5210.

14. Upon information and belief, Defendant Amazon Web Services, Inc. is a Delaware corporation with its headquarters located in the State of Washington at 410 Terry Ave. North, Seattle, WA 98109-5210.

SUBJECT MATTER JURISDICTION

15. This action arises under the Patent Act of the United States of America, 35 U.S.C. § 1 *et seq*.

16. This Court has subject matter jurisdiction pursuant to 28 U.S.C. §§ 1331 and 1338(a).

PERSONAL JURISDICTION

17. 35 U.S.C. § 271 provides, in pertinent part:

> (a) Except as otherwise provided in this title, whoever without authority makes, uses, offers to sell, or sells any patented invention, within the United States or imports into the United States any patented invention during the term of the patent therefor, infringes the patent.
>
> (b) Whoever actively induces infringement of a patent shall be liable as an infringer.
>
> (c) Whoever offers to sell or sells within the United States or imports into the United States a component of a patented machine, manufacture, combination or composition, or a material or apparatus for use in practicing a patented process, constituting a material part of the invention, knowing the same to be especially made or especially adapted for use in an infringement of such patent, and not a staple article or commodity of commerce suitable for substantial noninfringing use, shall be liable as a contributory infringer.

18. Both Amazon and AWS have offered and continue to offer infringing products and services for sale to Arizona residents and others in this judicial District.

19. Both Amazon and AWS maintain ongoing product development, sales teams and digital sales platforms, and customer support facilities for their infringing products and services in this judicial District.

20. Both Amazon and AWS have committed and continue to commit acts of infringement of the Patents-in-Suit in this judicial District.

21. Amazon and AWS maintain and operate an extensive web-based ecosystem at https://aws.amazon.com/ (the "AWS website").

22. Both Amazon and AWS, through the AWS website and other digital and physical facilities, regularly and systematically develop, manufacture, advertise, market, distribute, sell, offer for sale, and otherwise promote and support a variety of products and services which infringe the Patents-in-Suit.

23. Upon information and belief, on May 28, 2020, AWS filed its 2020 Annual Report with the Arizona Corporation Commission. A true and correct copy of a record for that submission as retrieved from the ACC database is attached hereto as Exhibit C. The 2020 Annual Report indicates that AWS is authorized to issue 1,000,000 shares of common stock, and that a total of one (1) share of that common stock has been issued.

24. According to an article entitled "AWS is Amazon's income generator, and more from the company's latest earnings" written by Jonathan Capriel and published by the Washington Business Journal on October 25, 2019 (Exhibit D):

> Amazon Web Services, meanwhile, Amazon's wildly profitable cloud computing subsidiary, registered operating income of $2.2 billion, or about 71% of the company's entire operating income.

25. The Amazon Defendants' annual revenues attributable to infringing products and services facilitated by the AWS website represent a significant and growing proportion of the Amazon Defendants' total annual revenues.

26. Upon information and belief, the Amazon Defendants employ thousands of full-time employees in Arizona and continue to actively recruit new employees in Arizona into positions involving software engineering, product development, project management, and extensive sales and support teams dedicated to AWS' cloud-based products and services.

27. Upon information and belief, many of these employees maintain home-based offices in Arizona, and store Amazon related materials (digitally and otherwise) at their home-based offices in this District, which materials can be distributed and/or offered for sale in this District.

28. Both Amazon and AWS maintain a physical place of business with an ongoing and continuous presence in this District.

29. Upon information and belief, both Amazon and AWS regularly conduct business including accepting orders, making business decisions, and soliciting customers in this District.

30. Upon information and belief, both Amazon and AWS have ratified or otherwise represented that they maintain a place of business in this District, for example by placing their names and/or logos on signage associated with their places of business.

31. The Amazon Defendants have conducted and continue to conduct business in this judicial District, including purposefully directing their actions and their online sales platform (the AWS website) into this judicial District and otherwise availing themselves of the privileges and protections of the laws of the State of Arizona.

32. The Amazon Defendants have sufficient minimum contacts within this District such that the exercise of jurisdiction over the Amazon Defendants by this Court does not offend traditional notions of fair play and substantial justice.

33. This Court has personal jurisdiction over the Amazon Defendants consistent with the principles of Due Process as embodied in the Fourteenth Amendment to the United States Constitution and the Arizona long-arm statute, Arizona Rule of Civil Procedure 4.2.

34. This action for patent infringement arises out of and relates to activities that the Amazon Defendants have purposefully directed into this District.

VENUE

35. 28 U.S.C. § 1400(b) provides that:

> Any civil action for patent infringement may be brought in the judicial district where the defendant resides, or where the defendant has committed acts of infringement and has a regular and established place of business.

36. Venue is proper in this District under 28 U.S.C. § 1400 because both Amazon Defendants, namely, Amazon and AWS, make, use, sell, and/or offer to sell infringing products and services in Arizona, either alone or in concert with each other and/or one or more additional entities as may be determined through discovery.

37. Venue is proper in this District because each Amazon Defendant also maintains regular and established places of business within Arizona.

38. Defendant Amazon maintains corporate offices in this judicial District at 222 South Mill Avenue, Tempe, Arizona 85281 (*See*, Exhibit E; https://www.downtowntempe.com/go/amazoncom).

39. According to an article written by Russ Wiles and published by the Arizona Republic August 18, 2020 entitled "Amazon to add 500 jobs at expanded tech hub in Tempe" (Exhibit F):

> Amazon.com, which already ranks as one of Arizona's largest employers, said it will hire 500 people over the next two years at an expanded technology hub near Tempe Town Lake.
>
> The online retailer said it will lease 90,000 square feet of space in a building under construction at 100 S. Mill Ave., part of the Hayden Ferry Lakeside development just north of the Arizona State University campus.
>
> The complex will feature 1.6 million square feet of office, retail and residential space. Amazon also has rented space in other nearby buildings, where it employs about 3,000 people.

(https://www.azcentral.com/story/money/business/jobs/2020/08/18/amazon-hiring-500-jobs-expanded-tech-center-tempe/3382413001/).

40. According to an article written by Erin Edgemon, Managing Editor, Phoenix Business Journal published October 21, 2020 and entitled "Amazon to hire 1,000 at new West Valley fulfillment center," Amazon operates a large fulfillment center at 16920 W Commerce Drive, Goodyear, Arizona 85338 (Exhibit G):

> Amazon has at least 11 new fulfillment and delivery operations sites set to open before the end of the year in Arizona. The e-commerce giant is in the process of hiring thousands in the region to operate them.

(https://www.bizjournals.com/phoenix/news/2020/10/21/amazon-to-hire-at-goodyear-fulfillment-center.html).

41. According to an article posted August 19, 2020, by the City of Goodyear in City News entitled "Amazon Becomes Goodyear's Largest Employer With New Robotics Facility - The 855,000-square-foot facility will add over 1,000 full-time jobs in Goodyear to its existing employee base" (Exhibit H):

> Amazon has been in Arizona over 13 years and in Goodyear for just over a decade. The company now has 17,500 full- and part-time employees in the state.
>
> "The talent and the labor force in Arizona have been a great partner over the last 13 years," said Matthew High, regional director of Amazon Fulfillment. "We've had a great relationship with the community, as well as the local officials, and we're excited to continue growing out here."

(https://www.goodyearaz.gov/Home/Components/News/News/11680/1549.

42. As of March 4, 2021, the Amazon Defendants' "Jobs" website (https://www.amazon.jobs/en) advertised a total of 370 (363 full-time) active job postings in Arizona (Exhibit I). These active listings specifically identify 170 full-time positions within AWS. (Exhibit J).

43. According to an article published March 14, 2019 and entitled "ASU, Amazon Web Services open Smart City Cloud Innovation Center," AWS launched its Cloud Innovation Center (CIC) Powered by AWS at ASU's Skysong location in Scottsdale, Arizona in March 2019 (Exhibit K).

44. According to an article dated January 20, 2021 entitled "ASU's Smart City Cloud Innovation Canter is 'working backwards' to innovate the future," AWS's Arizona-based Cloud Innovation Center (CIC) uses a product development approach referred to by Amazon as "working backwards" (Exhibit L).

45. In a Slideshare™ presentation dated April 25, 2018, AWS provided a tutorial on its "Working Backwards" methodology. (*See*, Exhibit M; https://www.slideshare.net/AmazonWebServices/working-backwards-from-the-customer).

46. Defendant Amazon has committed acts of infringement and maintains regular and established places of business in this District by, *inter alia*, operating extensive product development, fulfillment, software engineering, sales, manufacturing, distribution, and customer service organizations in Arizona.

47. Defendant AWS has committed acts of infringement and maintains regular and established places of business in this District by, *inter alia*, operating extensive product development, fulfillment, software engineering, sales, manufacturing, distribution, and customer service organizations in Arizona.

48. Plaintiff Swarm Technology, LLC is an Arizona LLC, and its principal place of business is in this District.

49. Alfonso and Alejandra Íñiguez reside in this District.

50. Swarm's products, design and development records, intellectual property, and related documents are located in this District.

<u>THE RELATIONSHIP BETWEEN AMAZON AND AWS</u>

51. Upon information and belief, AWS is a wholly owned subsidiary of Amazon.

52. Upon information and belief, Amazon operates AWS as an integrated business division of Amazon.

53. Upon information and belief, Amazon and AWS have a history of common officers and directors including, *inter alia*, Andrew Jassy, Peter DeSantis, and Stephen Schmidt.

54. Upon information and belief, the day-to-day operations of Amazon and AWS are coordinated and controlled by Amazon's executive suite, nicknamed the "S-team," which includes executives from both AWS and Amazon.

55. Amazon is traded on the NASDAQ Global Select Market Exchange under the symbol AMZN, for and on behalf of itself and its subsidiaries including AWS.

56. Upon information and belief, neither Amazon nor AWS are separately traded on the NASDAQ or any other exchange under any other symbol.

57. Amazon's "Notice of 2020 Annual Shareholders & Proxy Statement" ("Statement") dated May 27, 2020 (Exhibit N) states that:

> Sales by independent third-party sellers – mostly small and medium-sized business (SMBs) - make up more than half the units sold in our stores. In 2019, we invested $15 billion in infrastructure, services and tools, programs, and people to enable the success of these businesses.

58. The aforementioned May 27, 2020 Statement further explains that SMBs selling their products on Amazon "[u]se AWS to run their businesses" because "AWS offers low cost, on-demand IT solutions to help start-ups build and launch their applications and services."

59. Rather than promoting AWS as a separate company, Amazon offers its SMBs a seamless, integrated platform including on-line retail services under the Amazon brand, as well as cloud-based IT solutions under the AWS brand.

60. Upon information and belief, Amazon filed a single Annual Report (Form 10-K) with the Securities and Exchange Commission for the fiscal year ended December 31, 2020, on behalf of both Amazon and AWS. (Exhibit O).

61. Page 3 of Amazon's 2020 10-K explained that Amazon's operations are organized into three segments: North America, International, and AWS. Page 3 further stated that Amazon serves "developers and enterprises of all sizes, including start-ups, government agencies, and academic institutions, through AWS, which offers a broad set of on-demand technology services, including compute, storage, database, analytics, and machine learning, and other services." (*Id.*)

62. Amazon's 2020 10-K further provided selected consolidated financial data for Amazon and its various divisions (*e.g.*, AWS) including statements of cash flow, operations, income, balance sheets, and stockholder's equity, and did not provide separate financial statements for AWS.

63. The United States Trademark Office has issued a large number of federal trademark registrations for the words AMAZON, AMAZON.COM, AMAZON WEB SERVICES, INC. and AWS, both with and without design elements such as the grocery cart logo or the curved arrow underlying the word AMAZON and pointing from the letter "a" to the letter "z."

64. Many of the aforementioned trademarks involve goods and services spanning both: i) online retail/e-commerce products and services offered under the Amazon brand, as well as ii) IT on-demand products and services offered under the AWS brand.

65. Upon information and belief, all of the foregoing registrations are commonly owned by Amazon Technologies, Inc., a Nevada corporation doing business at 410 Terry Avenue North, Seattle, Washington, which is the same address where the headquarters of both Amazon and AWS are located.

66. The following exemplary registrations recite goods and services covering both online retail and IT on-demand goods/services, and include AMAZON or

AMAZON.COM but not "web services" or "AWS": 6228267; 3414814; 4171965; 5655933; 6019093; and 3411872. True and correct copies of the corresponding records retrieved from the USPTO's Trademark Electronic Search System (TESS) are attached hereto as Exhibit P.

67. The following registrations recite goods and services covering both online retail and IT on-demand goods/services, and include AMAZON WEB SERVICES or AWS but not "Amazon.com": 5676725; 6273100; 5878542; 1167503; and 3576161. True and correct copies of the corresponding records retrieved from the USPTO's Trademark Electronic Search System (TESS) are attached hereto as Exhibit Q.

68. Upon information and belief, Amazon exercises substantially total control over AWS or, alternatively, Amazon's and AWS's operations are commonly controlled or otherwise intermingled such that each entity's infringing acts and resulting liability may be imputed to the other.

69. Upon information and belief, Amazon exercises substantially total control over AWS or, alternatively, Amazon's and AWS's operations are commonly controlled or otherwise intermingled such that each entity's contacts within Arizona may be imputed to the other for purposes of personal jurisdiction.

THE STORY BEHIND MR. ÍÑIGUEZ' INVENTIONS

70. Alfonso Íñiguez was born in Tijuana, Mexico in 1965. He is pictured below (on the far right) with his mother and three siblings in approximately 1970:

71. Alfonso displayed remarkable abilities in science, technology, and mathematics at an early age. He also had an interest in how colonies of ants communicate with one another to perform tasks for the benefit of the queen, but without communicating with the queen directly. He would later apply his study of ant colonies to his design and development of co-processing architectures for computer systems.

72. While working at the American Consulate in Nogales, Mexico, Alfonso's mother obtained a United States Green Card. After leaving her employment at the Consulate in 1975, she submitted a Green Card application for Alfonso when he was ten (10) years old. Instilled with an impeccable work ethic, Alfonso went on to receive a Bachelor of Science degree in Computer Engineering from the Universidad Autonoma de Guadalajara, México in 1989.

73. Alfonso obtained his Green Card in 1987 and emigrated to the United States in 1989 to pursue graduate studies. While working full-time in various computer-related fields, Mr. Íñiguez attended the University of Arizona in Tucson, Arizona, and became a U.S. Citizen in 1994. In 1995 he was awarded a Master of Science degree in Electrical Engineering from the University of Arizona.

74. During the 2009 recession, Mr. Íñiguez was one of many employees laid off at Freescale Semiconductor (formerly Motorola, Inc.). After an extensive search, he secured an interview with a leading chip manufacturer as a Computer Architect.

75. Mr. Íñiguez prepared for his interview by reading books, papers, and performing extensive research in the field of computer architecture. He was struck by the inefficiencies associated with state-of-the-art computer processing architectures. He intuitively knew there was a better way for computer processors to cooperate with each other and with a central controller to perform complex processing tasks. He also believed that the swarm intelligence exhibited by ant colonies would play an important role in his new processing paradigm.

76. Drawing on his computer industry experience, Mr. Íñiguez identified two major drawbacks with existing multiprocessing frameworks. First, a significant portion of the CPU's processing cycles (bandwidth) was consumed assigning tasks to the co-processors. Second, the processors were often idle while waiting for a new task.

77. To address these shortcomings, Mr. Íñiguez invented a revolutionary new parallel processing paradigm, generally characterized by co-processors configured to proactively seek new tasks without having to communicate directly with (or wait for) the CPU. These co-processors include hardware and/or software components which are variously referred to as "autonomous agents" configured to retrieve "tasks" or "device shadows."

78. On January 25, 2013, Mr. Iñiguez filed his first utility patent application with the United States Patent and Trademark Office, and thereafter filed additional utility patent applications, each claiming priority to the original January 2013 filing date.

79. On September 29, 2015, the United States Patent and Trademark Office (the "USPTO") awarded U.S. Patent No. 9,146,777 entitled "Parallel Processing with Solidarity Cells by Proactively Retrieving from a Task Pool a Matching Task for the Solidarity Cell to Process" to Swarm.

80. On 26 December 2017, the USPTO awarded U.S. Patent No. 9,852,004 entitled "System and Method for Parallel Processing using Dynamically Configurable Proactive Co-Processing Cells" to Swarm.

81. On 17 March 2020, the USPTO awarded U.S. Patent No. 10,592,275 entitled "System and Method for Swarm Collaborative Intelligence using Dynamically Configurable Proactive Autonomous Agents" to Swarm.

82. Swarm is the sole owner of all right, title, and interest in and to each of the foregoing Patents-in-Suit.

83. Various products and services made, used, sold, offered for sale, or imported into the Unites States by the Amazon Defendants embody every element of at least one claim of each Patents-in-Suit, whether directly, contributorily, and/or through inducement (35 U.S.C. § 271), either literally or under the doctrine of equivalents.

84. The Patents-in-Suit disclose several embodiments, including a processing system having a controller configured to populate a task pool and one or more co-processors configured to proactively retrieve tasks from the task pool. In this way, the controller communicates directly with the task pool, and indirectly with the co-processors through the task pool.

85. Mr. Iñiguez contemplated many practical applications of his inventions, one of which included networks comprising Internet of Things (IoT) devices. One problem faced by engineers and computer architects surrounds the control of large numbers of devices linked to an IoT network, and how to harness their collective processing capacity without over-burdening the CPU.

86. The demand for IoT devices and IoT networks continues to drive growth in cloud-based products and services involving computing, storage, networking, databases, analytics, application services, deployment, mobile tools, and developer tools. Present day IoT networks make these services available to virtually any device connected to the Internet.

87. To help explain the technology, Mr. Íñiguez and his family created a series of homemade videos, each lasting approximately four minutes. The videos feature teams of ants working together and completing tasks, giving a visual metaphor of his revolutionary new computer architecture. These videos may be viewed at:

https://vimeo.com/150745850 (CPU Architecture - Prior Art)

https://vimeo.com/150748582 (Co-Processor Architecture - Prior Art)

https://vimeo.com/150450660 (Solidarity Cell Architecture - Issued Patent)

https://vimeo.com/150743111 (Heterogeneous Parallel Processing - Issued Patent)

https://vimeo.com/150759740 (Internet of Things Parallel Processing - Issued Patent)

https://vimeo.com/150744874 (Robot Automation - Issued Patent)

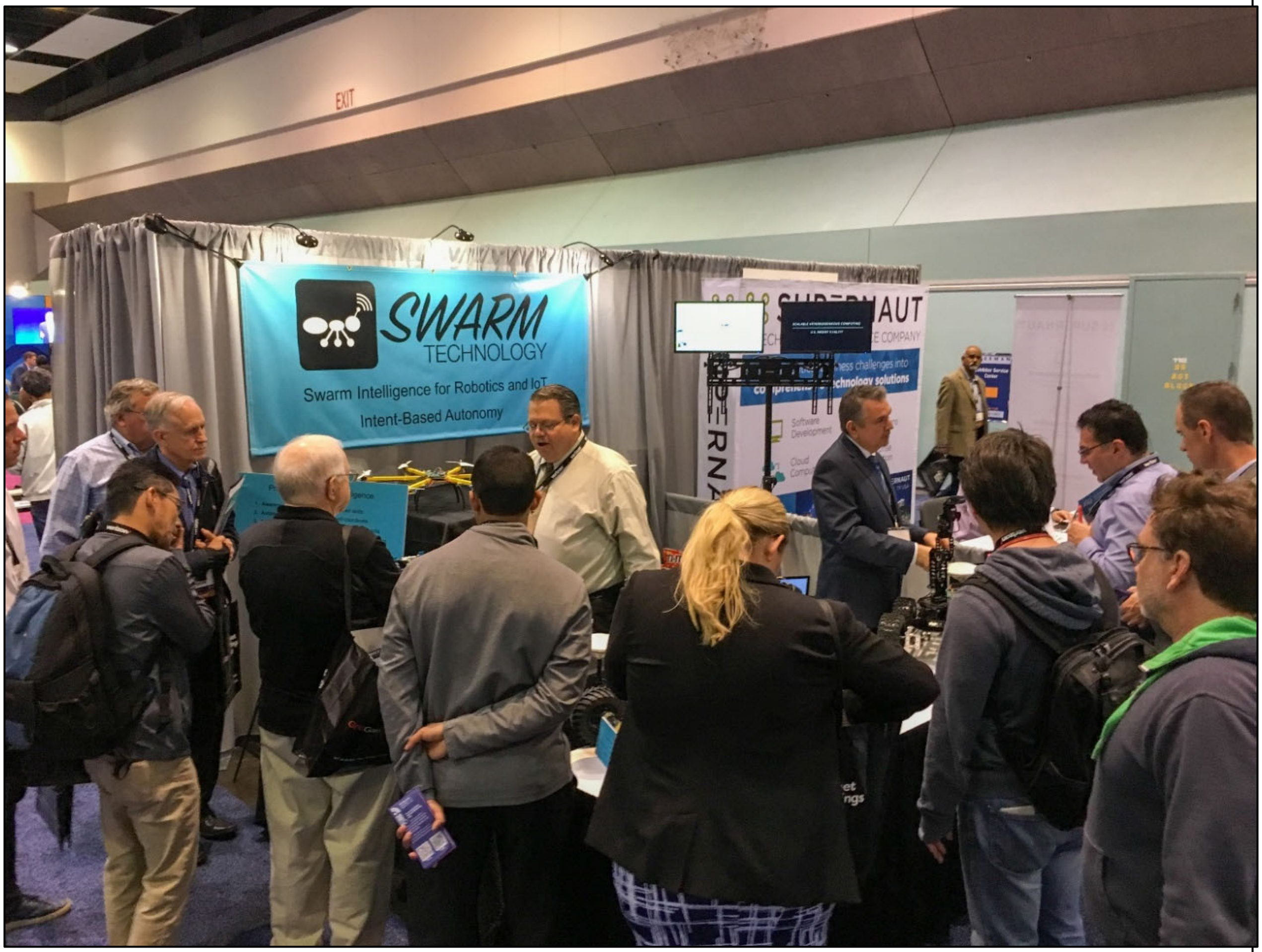
https://vimeo.com/177881911 (Ants, Robots, IoT and Parallel Processing)

88. Mr. Íñiguez and his family presented his technology at trade shows and other industry events, such as the: i) "Internet of Things World Conference 2018," Santa Barbara California, May 14 – 17, 2018; ii) "IoT Tech Expo North America 2017," Santa Clara, California, November 29-30, 2017; iii) "International Conference on Intelligent Robots and Systems (IROS) 2017," Vancouver, Canada, September 24–28, 2017; and iv) "Internet of Things World Conference 2017," Santa Clara, California, May 16-18, 2017.

89. Below is a photograph (left-to-right) of the Íñiguez family including sons Ulises and Isaac, daughter Daniela, wife Alejandra, and husband Alfonso promoting Swarm at an industry event in 2017:

90. Ulises Íñiguez recently graduated from the University of Notre Dame with a B.S. in Mechanical Engineering. Isaac Íñiguez is currently a Senior at Franciscan University studying Business and Marketing. Daniela Íñiguez is currently a Sophomore at Franciscan University studying Engineering (scheduled to transfer to the University of Notre Dame to complete a B.S. in Mechanical Engineering). All three Íñiguez children have studied on academic scholarships; currently, Isaac and Daniela are studying on academic scholarships.

91. Below is a photograph of Alfonso Íñiguez (right) and his cousin Pablo Garcia (B.S. Industrial Engineering - Instituto Tecnológico de Sonora, Mexico) promoting Swarm at an industry event in 2018:



92. Mr. Íñiguez 's technology has also been the subject of news articles and other press coverage, such as the IEEE News in May of 2017, the Business News in April of 2018, the East Valley Tribune in April 2016, the Business Journal in December of 2015, and the EE Times in December of 2017, among others.

93. Mr. Íñiguez is also the author of the following peer reviewed research paper entitled: "The Octopus as a Model for Artificial Intelligence: A Multi-Agent Robotic Case Study." The paper was published by the International Conference on Agents and Artificial Intelligence held in Porto, Portugal, in 2017. The International Conference on Agents and Artificial Intelligence is the most prestigious Artificial Intelligence conference in the World. It is extremely rare to include a company researcher (as opposed to a university

researcher) as a featured author. A true and correct copy of the paper is attached hereto as Exhibit R and may be found at: https://www.scitepress.org/Papers/2017/61254/61254.pdf.

94. Mr. Íñiguez recently discovered that the many technology companies have incorporated his technology into their own products and services and are marketing them to their own customers. Mr. Íñiguez determined that at least the following Amazon products and services infringe the Patents-in-Suit: i) AWS IoT Core; and ii) AWS IoT Greengrass (the "Infringing Products"). Product literature promoting and offering these services for sale in Arizona may be viewed at:

https://aws.amazon.com/iot-core/?hp=tile&so-exp=below and

https://aws.amazon.com/greengrass/?hp=tile&so-exp=below,

respectively.

95. After Mr. Íñiguez's first patent issued in September 2015, he contacted Amazon in an effort to initiate licensing discussions.

96. Specifically, on November 23, 2015, Swarm's outside patent counsel sent a letter to Mr. Werner Vogels at Amazon announcing the issuance of U.S. Patent No. 9,146,777 (the "'777 Patent"), and the pendency of additional related patent applications. A true and correct copy of the 2015 letter is attached hereto as Exhibit S.

97. On February 10, 2016, Mr. Íñiguez sent an email to Amazon at patents@amazon.com describing Swarm's technology. A true and correct copy of the February 2016 email is attached hereto as Exhibit T.

98. On August 10, 2016, Mr. Íñiguez sent an email to Amazon Robotics at patents@amazon.com including links to Swarm's videos. A true and correct copy of the August 2016 email is attached hereto as Exhibit U.

99. On July 16, 2018, Swarm's outside patent counsel sent a letter to Jeffrey Blackburn at Amazon offering a license under the '777 Patent, the '004 Patent, and the pending application No. 15/852,480 which later issued as the '275 Patent. A true and correct copy of the 2018 letter is attached hereto as Exhibit V.

100. The Amazon Defendants were provided written notice of the Patents-in-Suit at least four times between 2015 and 2018. As such, their infringement may be willful under 35 U.S.C. § 284.

101. The '004 Patent describes a system and method for parallel processing using dynamically configurable proactive co-processing cells.

102. One embodiment described in the '004 Patent includes a controller configured to populate a task pool. One or more co-processors, or solidarity cells, are configured to proactively retrieve tasks from the task pool. Each co-processor includes an agent that interrogates the task pool to seek a task to perform. As a result, the co-processors work together "in solidarity" with each other and with the task pool. Each co-processor is autonomous in that it may interact with the task pool without being instructed to do so by the controller. Devices and their associated co-processors may be added to a network on a "plug and play" basis.

103. A representative figure is below:

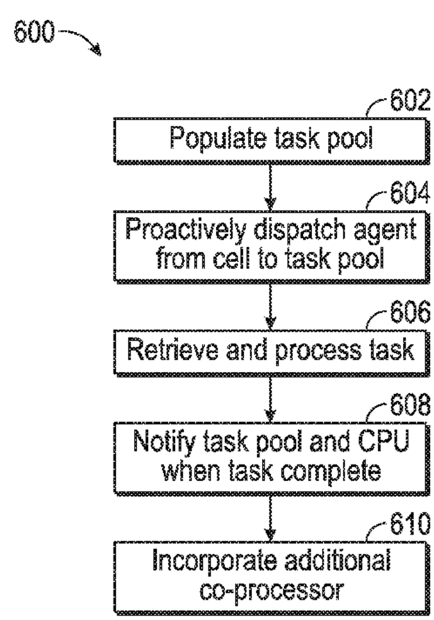


FIG. 6

104. The controller populates the task pool (602). The co-processors proactively dispatch an agent to the task pool (604). Each co-processor retrieves and processes a task (606). The task pool and controller are notified when the task is complete (608). An additional co-processor may be incorporated, "on boarded," or provisioned (610).

105. Claim 1 of the '004 Patent relates to a processing system including a task pool, a controller, and first and a second co-processors each configured to retrieve tasks from the task pool and update the task pool once the task is processed, without requiring direct communication with the controller.

106. Claim 1 of the '004 Patent is set forth in its entirety below:

> 1. A processing system, comprising:
>
> a task pool;
>
> a controller configured to populate the task pool with a plurality of first tasks and a plurality of second tasks;
>
> a first co-processor configured to successively: retrieve a first task from the task pool; deliver the first task to the first co-processor; process the first task; generate first resulting data; and update the task pool to reflect completion of the first task, all without any communication between the first co-processor and the controller; and
>
> a second co-processor configured to successively: retrieve a second task from the task pool; deliver the second task to the second co-processor; process the second task; generate second resulting data; and update the task pool to reflect completion of the second task, all without any communication between the second co-processor and the controller;
>
> wherein the processing system is configured to dynamically accept the first co-processor, the second co-processor, and an additional co-processor into the processing system on a plug-and-play basis without any communication with the controller.

107. As discussed below in conjunction with publicly available literature published in Arizona, many of the Amazon Defendants' products and services embody all of the foregoing elements of claim 1, as well as claims 2 – 12 of the '004 Patent.

108. As a result of the Amazon Defendants' infringement of the '004 Patent, Swarm has incurred substantial monetary and other damages.

109. The '275 Patent describes a system and method for collaborative intelligence using dynamically configurable proactive autonomous agents.

110. Claim 1 of the '275 Patent relates to a collaborative intelligence system including a task pool, a controller configured to populate the task pool with a plurality of tasks, and first and a second co-processors each configured to retrieve tasks from the task pool and update the task pool to reflect completion of the task, without requiring direct communication with the controller, and to autonomously work together in solidarity with the task pool to complete a common objective.

111. Claim 1 of the '275 Patent is set forth in its entirety below

> 1. A collaborative intelligence system, comprising:
>
> a task pool;
>
> a controller configured to populate the task pool with a plurality of first tasks and a plurality of second tasks;
>
> a first co-processor configured to successively: proactively retrieve a first task from the task pool; process the first task; generate first resulting data; and update the task pool to reflect completion of the first task, all without any communication between the first co-processor and the controller; and
>
> a second co-processor configured to successively: proactively retrieve a second task from the task pool; process the second task; generate second resulting data; and update the task pool to reflect completion of the second task, all without any communication between the second co-processor and the controller;
>
> wherein the collaborative intelligence system is configured to dynamically accept the first co-processor, the second co-

> processor, and an additional co-processor into the processing system on a plug-and-play basis without any communication with the controller;
>
> the plurality of first tasks and the plurality of second tasks are associated with a common objective;
>
> the first and second co-processors autonomously work together in solidarity with the task pool to complete the common objective.

112. As detailed below in conjunction with publicly available literature published in Arizona, many of the Amazon Defendants' products and services embody all of the foregoing elements of claim 1, as well as claims 2 – 17 of the '275 Patent.

113. As a result of the Amazon Defendants' infringement of the '275 Patent, Swarm has incurred substantial monetary and other damages.

114. The '777 Patent describes an apparatus for parallel processing of a large computing requirement.

115. Claim 1 of the '777 Patent includes a CPU, a task pool, and a solidarity cell having agent configured to proactively retrieve a matching task from the task pool, without requiring an instruction from the CPU.

116. Claim 1 of the '777 Patent is set forth in its entirety below:

> 1. An apparatus for parallel processing of a large computing requirement, the apparatus comprising:
>
> a central processing unit ("CPU");
>
> a task pool in electronic communication with the CPU; and
>
> a first solidarity cell in electronic communication with the task pool, the first solidarity cell comprising a first agent configured to proactively retrieve, from the task pool, without requiring an instruction from the CPU, a matching task for the solidarity cell to process;
>
> wherein the CPU populates the task pool by dividing the requirement into one or more threads and placing the threads in the task pool, each thread comprising one

or more tasks, and the matching task being one of the tasks;

wherein each task comprises a descriptor, the descriptor containing at least:

a function to be executed; and

a memory location of data upon which the function is to be executed;

wherein the first agent is a data frame comprising:

a source address, a destination address and a payload;

wherein the first agent retrieves the matching task by:

being dispatched by the first solidarity cell to the task pool, during which the source address is the first solidarity cell's address, the destination address is the task pool's address, and the payload comprises a list of functions the first solidarity cell is configured to perform;

searching the task pool for a task that is ready to be processed and has a function that the first solidarity cell can perform; and

returning to the first solidarity cell, during which the source address is the task pool's address, the destination address is the first solidarity cell's address, and the payload comprises the descriptor of the matching task.

117. As detailed below in conjunction with publicly available literature published in Arizona, many of the Amazon Defendants' products and services embody all of the elements of claim 1, as well as claims 2 - 14 of the '777 Patent.

118. As a result of the Amazon Defendants' infringement of the '777 Patent, Swarm has incurred substantial monetary and other damages.

119. The Amazon Defendants are building their future on the back of Mr. Íñiguez' novel computer architecture. The widely recognized problem of controlling multiple IoT

devices has been solved by Alfonso Íñiguez. The Patents-in-Suit directly address many of the challenges faced by today's software developers, and the Amazon Defendants know this.

120. 35 U.S.C. § 271(a) provides that whoever "makes, uses, offers to sell, or sells any patented invention, within the United States or imports into the United States any patented invention" infringes the patent. As described below, Exhibits X and Z include detailed claim charts demonstrating the direct infringement of the Patents-in-Suit.

121. 35 U.S.C. § 271(b) provides that "[w]hoever actively induces infringement of a patent shall be liable as an infringer." Inducement often involves a showing that the alleged inducer knew of the patent, knowingly induced the infringing acts, and possessed a specific intent to encourage another's infringement of the patent. As described below, Amazon was either aware of or willfully blind to the Patents-in-Suit, for example as a result of pre-suit correspondence between Swarm and Amazon. At a minimum, Amazon has known of the Patents-in-Suit since at least as early as March 15, 2021 (the date Swarm filed its initial complaint in this District). Moreover, Amazon has specifically encouraged others to participate in the infringement, as evidenced by Amazon's published product descriptions and specifications, discussed below and shown in Exhibits W, X, and Z; that is, Amazon has instructed others how to use Amazon's products and services in an infringing manner.

122. 35 U.S.C. § 271(b) provides that whoever "offers to sell or sells within the United States or imports into the United States a component of a patented machine, manufacture, combination or composition, or a material or apparatus for use in practicing a patented process, constituting a material part of the invention, knowing the same to be especially made or especially adapted for use in an infringement of such

patent, and not a staple article or commodity of commerce suitable for substantial noninfringing use, shall be liable as a contributory infringer."

123. Upon information and belief, early discovery will reveal facts and circumstances confirming that Amazon and others made, used, sold, or offered for sale at least a material part of Swarm's inventions knowing that they would be used in the Infringing Products. Moreover, Amazon's detailed product literature evidences a specific intent to encourage others to participate in the infringement of Swarms Patents.

THE '004 PATENT

I. SWARM INVENTED NEW AND USEFUL PROCESSES AND MACHINES WHICH IMPROVE THE OPERATION OF A COMPUTER AND WHICH INCLUDE INVENTIVE CONCEPTS.

124. Claim 1 of the '004 Patent is set forth below in its entirety:

A processing system, comprising:

a task pool;

a controller configured to populate the task pool with a plurality of first tasks and a plurality of second tasks;

a first co-processor configured to successively: retrieve a first task from the task pool; deliver the first task to the first co-processor; process the first task; generate first resulting data; and update the task pool to reflect completion of the first task, all without any communication between the first co-processor and the controller; and

a second co-processor configured to successively: retrieve a second task from the task pool; deliver the second task to the second co-processor; process the second task; generate second resulting data; and update the task pool to reflect completion of the second task, all without any communication between the second co-processor and the controller;

wherein the processing system is configured to dynamically accept the first co-processor, the second co-processor, and an additional co-processor into the

> processing system on a plug-and-play basis without any communication with the controller.

A. **Swarm Invented a New Processing Architecture.**

125. The preamble of Claim 1 recites:

> A processing system, comprising: ('004; 14:10).

126. The '004 specification describes various processing systems, for example in the context of:

> [A] a distributed processing system 10 includes a single or multi-core CPU 11 and one or more solidarity or co-processing cells 12A-12 configured to communicate with a task pool 13 ... ('004; 4:31-34); and

> [A] parallel processing computer architecture [including] a CPU configured to populate a task pool, and one or more co-processors configured to proactively retrieve threads (tasks) from the task pool. ('004; 2:11-14).

127. The claimed processing system involves new and useful machines and processes, and new and useful improvements to machines and processes. Taken together, the controller, task pool, and co-processors confer a substantial advantage over prior art processing systems by allowing different types of co-processors to interact with the task pool without significantly compromising their individual performance. (*See Visual Memory LLC v. NVIDIA Corporation*, 867 F.3d 1253, 1256-1257 (Fed. Cir. 2017)). Claim 1 thus focuses on improvements to computer functionality, as opposed to merely being directed to an abstract idea. (*See Enfish, LLC v. Microsoft Corp.,* 822 F.3d 1327, 1336 (Fed. Cir. 2016)). Moreover, these improvements are agnostic to the *type* of activities (whether human or non-human) to be processed.

128. Even assuming, *arguendo*, that claim 1 is directed to an abstract idea involving organizing human activities such as a scrum board (it is not), claim 1 nonetheless includes inventive concepts that amount to significantly more than an abstract idea. For example, each co-processor may be configured to retrieve a task by sending its agent to the

task pool when the co-processor is idle or otherwise able to contribute processing cycles without impeding its normal operation. In this context, the term agent refers to a software module, analogous to a network packet, associated with a co-processor that interacts with the task pool to obtain tasks which are appropriate for that co-processor. ('004; 3:1-14). Humans are not capable of performing tasks such as transmitting a network packet from a co-processor to a data structure (*e.g.*, task pool); they are specific to computer operation.

B. <u>Swarm Invented a New Processing Architecture Comprising a Task Pool Interposed Between the CPU and the Co-Processors.</u>

129. Claim 1 further recites:

> a task pool ('004; 14:11).

130. The '004 specification describes the new processing architecture in terms of the interaction among the task pool, the controller (CPU), and the co-processors:

> The co-processors may also be capable of acting autonomously; that is, they may interact with the task pool independently of the CPU. In a preferred embodiment, each co-processor includes an agent that interrogates the task pool to seek a task to perform. As a result, the co-processors work together "in solidarity" with one another and with the task pool to complete aggregate computational requirements by autonomously retrieving and completing individual tasks which may or may not be inter-related. ('004; 2:20- 28).

131. The task pool improves the operation of a computer by acting as an intermediary device between the CPU and the co-processors. More particularly, conventional processors include a CPU and one or more co-processors, where "[t]he CPU partitions the computational requirements into tasks and distributes the tasks to co-processors." ('004; 1:56-59). Consequently, "a significant amount of CPU bandwidth is consumed by task distribution; waiting for tasks to be completed before distributing new tasks (often with dependencies on previous tasks); responding to interrupts from co-processors when a task is completed; and responding to other messages from co-processors." ('004 1:63-2:1).

132. To address these shortcomings, Swarm invented a revolutionary new parallel processing paradigm, including co-processors configured to proactively retrieve new tasks from the task pool without having to communicate directly with (or wait for) the CPU.

133. Even assuming, *arguendo*, that claim 1 is directed to an abstract idea (it is not), claim 1 nonetheless includes inventive concepts involving more than well-understood, routine, and conventional activities previously known to the industry. (*See Aatrix Software, Inc. v. Green Shades Software, Inc.*, 882 F.3d 1121, 1128 (Fed. Cir. 2018)). For example, the CPU may be programmed "to recognize and communicate with the task pool 13 and divide the computing requirements into threads ...." ('004; 5:40-43). As a result, "a co-processor may interact with the task pool without being instructed to do so by the CPU or by the task pool" ('004; 2:38-40).

C. <u>Swarm Invented a New Processing Architecture Including a Controller Configured to Place Tasks Into the Task Pool.</u>

134. Claim 1 further recites:

> a controller configured to populate the task pool with a plurality of first tasks and a plurality of second tasks ('004; 14:12-13).

135. The '004 specification describes various controllers (CPUs), for example in the context of the multi-processor networks illustrated in FIGS. 1 and 4:

> Referring now to FIG. 4, an internet of things network 400 includes a controller (CPU) 402, a task pool 408, and various devices 410-422, some or all of which include an associated or embedded microcontroller, such as an integrated circuit (IC) chip or other component which embodies processing capacity. ('004; 11:35-40);
>
> ...
>
> In the illustrated embodiment, the controller 402 may be a smartphone, tablet, laptop, or other device which may include a display 404 and a user interface (*e.g.*, keypad) 406 for facilitating user interaction with the various devices on the network. ('004; 11:46-50);
>
> ...

> For example, in FIG. 1, the system 10 may divides an aggregate computational problem into a group of tasks, and populate the task pool 13 with a first type, a second type, and a third type of tasks. ('004, 6:39-42)

136. Claim 1 is directed to improvements to computer functionality because the controller's operating code is specifically programmed to cause the controller to distribute tasks to a task pool, as opposed to conventional processing systems in which the controller distributes tasks directly to the co-processors.

137. Even assuming, *arguendo*, that claim 1 is directed to an abstract idea (it is not), claim 1 nonetheless includes inventive concepts involving more than well-understood, routine, and conventional activities previously known to the industry. For example, "the CPU 11 may be configured for use within the system 10 by programming it to recognize and communicate with the task pool 13 and divide the computing requirements into threads." ('004, 5:40-43). By using the task pool as an intermediary device between the controller and the co-processors, the elements of claim 1, both individually and as a combination, specifically prevent and override the routine and conventional sequence of events performed by prior art processing architectures. (*See SRI Int'l, Inc. v. Cisco Sys., Inc.*, 918 F.3d 1368 (Fed. Cir. 2019) (quoting *DDR Holdings, LLC v. Hotels.com, L.P.,* 773 F.3d 1245, 1257 (Fed. Cir. 2014))).

D. **Swarm Invented a New Processing Architecture Comprising First and Second Co-Processors, Each Configured to Retrieve Tasks From the Task Pool Rather Than From the CPU.**

138. Claim 1 further recites:

> a first co-processor configured to successively: retrieve a first task from the task pool; deliver the first task to the first co-processor; process the first task; generate first resulting data; and update the task pool to reflect completion of the first task, all without any communication between the first co-processor and the controller ('004; 14:14-19); and
>
> a second co-processor configured to successively: retrieve a second task from the task pool; deliver the second task to the second co-processor; process the second task; generate second

> resulting data; and update the task pool to reflect completion of the second task, all without any communication between the second co-processor and the controller ('004; 14:21-27).

139. The '004 specification describes the configuration and operation of the first and second co-processors:

> Various embodiments of a parallel processing computing architecture include a CPU configured to populate a task pool, and one or more co-processors configured to proactively retrieve threads (tasks) from the task pool. Each co-processor notifies the task pool upon completion of a task, and pings the task pool until another task becomes available for processing. In this way, the CPU communicates directly with the task pool, and communicates indirectly with the co-processors through the task pool. ('004; 2:20- 28);
>
> ...
>
> Upon retrieving a task from the task pool, a cell may then process that task, typically by retrieving data from a particular location in first memory 304, processing that data, and storing the processed data at a particular location within second memory 306. When a task is completed, the cell notifies the task pool, the task pool marks the task as completed, and the task pool notifies the CPU that the task is completed. ('004; 11:21-28);
>
> ...
>
> Significantly, the retrieval of tasks and the processing of data by the cells may occur without direct communication between the CPU and the various cells. ('004; 11:31-34).

140. The first and second co-processors, both individually and in combination with each other and/or one or more additional co-processors, improve the operation of a computer by retrieving tasks from a task pool (rather than from the CPU). The co-processors further improve the operation of computers by updating the task pool to reflect task completion, as opposed to conventional processing architectures in which the co-processors directly update the CPU.

141. Even assuming, *arguendo*, that claim 1 is directed to an abstract idea (it is not), claim 1 nonetheless includes numerous inventive concepts. For example, the first and

second co-processors are specifically programmed to retrieve respective tasks from the task pool, and subsequently update the task pool after completing their respective tasks, without directly communicating with the controller.

142. Moreover, the specification refers to the co-processors as autonomous, proactive solidarity cells. In this context, the term autonomous implies that a co-processor may interact with the task pool without being instructed to do so by the CPU or by the task pool. The term proactive suggests that each co-processor may be configured (*e.g.*, programmed) to periodically send an agent to monitor the task pool for available tasks appropriate to that co-processor. The term solidarity implies that co-processing cells share a common objective in monitoring and executing all available tasks within the task pool. Prior to Swarm's invention, these inventive concepts had never been proposed before, and thus they involve more than well-understood, routine, and conventional activities previously known to the industry.

E. <u>**Swarm Invented a New Processing Architecture Configured to Dynamically Accept the First, Second, and an Additional Co-Processor on a Plug-and-Play Basis.**</u>

143. Claim 1 further recites:

> wherein the processing system is configured to dynamically accept the first co-processor, the second co-processor, and an additional co-processor into the processing system on a plug-and-play basis without any communication with the controller. ('004; 14:28-32).

144. The '004 specification describes the dynamic plug-and-play feature of the invention:

> [I]nteroperability among the CPU and co-processors may be facilitated by configuring the CPU to compose and/or structure tasks at a level of abstraction which is independent of the instruction set architecture associated with the various co-processors, thereby allowing the components to communicate at a task level rather than at an instruction level. As such, devices and their associated co-processors may be added to a network on a 'plug and play' basis. ('004; 3:34-40).

145. Dynamically accepting co-processors on a plug-and-play basis improves the operation of a computer network by integrating co-processors with different instruction set architectures into the same network. ('004; 3:43-44).

146. Even assuming, *arguendo*, that claim 1 is directed to an abstract idea (it is not), claim 1 nonetheless includes numerous inventive concepts. For example, the system may include a plurality of cells, wherein some of the cells are capable of performing the same task types as other cells, to thereby create redundancy in the system. This redundancy allows the system to continue functioning seamlessly when cells are removed from the system or are otherwise unavailable. The system also functions seamlessly when cells are dynamically added to the system. ('004; 6:34-55). These inventive concepts had never been proposed before Swarm invented them.

147. Accordingly, claim 1 is directed to a new processing architecture which improves the operation of computer, and which includes significantly more than well-understood, routine, and conventional activities.

148. Claims 2 – 12 of the '004 Patent are also directed to various features of a new processing architecture which improve the operation of computer, and which include significantly more than well-understood, routine, and conventional activities.

149. By way of non-limiting example, claim 4 is directed to a processing system "wherein the controller and the task pool reside on a *monolithic integrated circuit* (IC), and the first and second co-processors do not reside on the IC."

150. Claim 5 is directed to a processing system "wherein the controller, the task pool, and the first and second co-processors reside on a single monolithic integrated circuit (IC)."

151. Claim 6 is directed to a processing system "wherein the first and second devices each comprise one of a sensor, light bulb, power switch, appliance, biometric

device, medical device, diagnostic device, lap top, tablet, smartphone, motor controller, and security device."

152. Claim 7 is directed to a processing system "wherein the first co-processor is configured to modify a task within the task pool."

153. Claim 8 is directed to a processing system wherein "the first co-processor is further configured to process the first and notify the task pool upon completion of the first task."

154. Claim 9 is directed to a processing system wherein "the task pool is configured to notify the controller upon completion of the first task."

155. Claim 10 is directed to a processing system "wherein the controller is configured to communicate with the first co-processor and the second co-processor only indirectly through the task pool."

156. Claim 11 is directed to a processing system "wherein the first co-processor is configured to deposit a new task into the task pool."

157. Claim 12 is directed to a processing system "wherein the first co-processor is configured to determine when it has available processing capacity, and to dispatch the first agent to the task pool in response to the determination."

158. As explained in detail in the '004 specification, each of the foregoing claims are directed to improvements to the operation of computer, and include significantly more than well-understood, routine, and conventional activities.

<u>THE '275 PATENT</u>

II. <u>SWARM INVENTED NEW AND USEFUL PROCESSES AND MACHINES WHICH IMPROVE THE OPERATION OF A COMPUTER AND WHICH INCLUDE INVENTIVE CONCEPTS.</u>

159. Claim 1 of the '275 Patent is set forth below in its entirety:

A collaborative intelligence system, comprising:

a task pool;

> a controller configured to populate the task pool with a plurality of first tasks and a plurality of second tasks;
>
> a first co-processor configured to successively: proactively retrieve a first task from the task pool; process the first task; generate first resulting data; and update the task pool to reflect completion of the first task, all without any communication between the first co-processor and the controller; and
>
> a second co-processor configured to successively: proactively retrieve a second task from the task pool; process the second task; generate second resulting data; and update the task pool to reflect completion of the second task, all without any communication between the second co-processor and the controller;
>
> wherein the collaborative intelligence system is configured to dynamically accept the first co-processor, the second co-processor, and an additional co-processor into the processing system on a plug-and-play basis without any communication with the controller;
>
> the plurality of first tasks and the plurality of second tasks are associated with a common objective;
>
> the first and second co-processors autonomously work together in solidarity with the task pool to complete the common objective.

A. <u>Swarm Invented a New Processing Architecture.</u>

160. The preamble of Claim 1 recites:

> A collaborative intelligence system, comprising: ('275; 14:24).

161. The '275 specification describes various collaborative intelligence systems, for example in the context of:

> [P]arallel processing computing systems and environments (such as IoT and collaborative intelligence environments), ranging from simple switching and control functions to complex programs and algorithms including, without limitation: robot control, data encryption; graphics, video, and audio processing; direct memory access; mathematical computations; data mining; game algorithms; ethernet packet and other network protocol processing including construction, reception and transmission of data the outside network;

> financial services and business methods; search engines; internet data streaming and other web-based applications; execution of internal or external software programs; switching on and off and/or otherwise controlling or manipulating appliances, light bulbs, consumer electronics, robotic vehicles, and the like, *e.g.*, in the context of the Internet-of-Things and/or collaborative intelligence systems. ('275; 4:18-34).

162. The claimed collaborative intelligence system involves new and useful machines and processes, and new and useful improvements to machines and processes. Taken together, the controller, task pool, and co-processors confer a substantial advantage over prior art processing systems by allowing different types of co-processors to interact with the task pool without significantly compromising their individual performance. Claim 1 is thus directed to improvements to computer functionality, as opposed to merely being directed to an abstract idea. Moreover, these improvements are agnostic to the *type* of activities (whether human or non-human) to be processed.

163. Even assuming, *arguendo*, that claim 1 is directed to an abstract idea involving organizing human activities such as a scrum board (it is not), claim 1 nonetheless includes inventive concepts that amount to significantly more than an abstract idea. For example, each co-processor may be configured to retrieve a task by sending its agent to the task pool when the co-processor is idle or otherwise able to contribute processing cycles without impeding its normal operation. In this context, the term agent refers to a software module, analogous to a network packet, associated with a co-processor that interacts with the task pool to obtain tasks which are appropriate for that co-processor. ('275; 3:21-24). Humans are not capable of performing tasks such as transmitting a network packet from a co-processor to a data structure (*e.g.*, task pool), as they are specific to computer operation.

B. <u>Swarm Invented a New Processing Architecture Comprising a Task Pool Interposed Between the CPU and the Co-Processors.</u>

164. Claim 1 further recites:

> a task pool ('275; 14:25).

165. The ’275 specification describes the new processing architecture in terms of the interaction among the task pool, the controller (CPU), and the co-processors:

> The co-processors may also be capable of acting autonomously; that is, they may interact with the task pool independently of the CPU. In a preferred embodiment, each co-processor includes an agent that interrogates the task pool to seek a task to perform. As a result, the co-processors work together "in solidarity" with one another and with the task pool to complete aggregate computational requirements by autonomously retrieving and completing individual tasks which may or may not be inter-related. (’275; 2:28-36).

166. The task pool improves the operation of a computer by electronically communicating with the CPU as well as the co-processors. More particularly, conventional processors include a CPU and one or more co-processors, where "[t]he CPU partitions the computational requirements into tasks and distributes the tasks to co-processors." (’275; 1:63-64). Consequently, "a significant amount of CPU bandwidth is consumed by task distribution; waiting for tasks to be completed before distributing new tasks (often with dependencies on previous tasks); responding to interrupts from co-processors when a task is completed; and responding to other messages from co-processors." (’275 2:3-8).

167. To address these shortcomings, Swarm invented a revolutionary new parallel processing paradigm, including co-processors configured to proactively retrieve new tasks from the task pool without having to communicate directly with (or wait for) the CPU.

168. Even assuming, *arguendo*, that claim 1 is directed to an abstract idea (it is not), claim 1 nonetheless includes inventive concepts involving more than well-understood, routine, and conventional activities previously known to the industry. For example, the CPU may be programmed "to recognize and communicate with the task pool 13 and divide the computing requirements into threads … ." (’275; 5:54-56). As a result, "a co-processor may interact with the task pool without being instructed to do so by the CPU or by the task pool" (’275; 2:46-48).

C. Swarm Invented a New Processing Architecture Comprising a Controller Configured to Place Tasks Into the Task Pool.

169. Claim 1 further recites:

> a controller configured to populate the task pool with a plurality of first tasks and a plurality of second tasks ('275; 14:26-27).

170. The '275 specification describes various controllers (CPUs), for example in the context of the multi-processor networks illustrated in FIGS. 1 and 4:

> Referring now to FIG. 4, an internet of things network 400 includes a controller (CPU) 402, a task pool 408, and various devices 410-422, some or all of which include an associated or embedded microcontroller, such as an integrated circuit (IC) chip or other component which embodies processing capacity. ('275; 11:51-56);
>
> ...
>
> In the illustrated embodiment, the controller 402 may be a smartphone, tablet, laptop, or other device which may include a display 404 and a user interface (*e.g.*, keypad) 406 for facilitating user interaction with the various devices on the network. ('275; 11:62-66);
>
> ...
>
> For example, in FIG. 1, the system 10 may divides an aggregate computational problem into a group of tasks, and populate the task pool 13 with a first type, a second type, and a third type of tasks. ('275, 6:54-57).

171. Claim 1 is directed to improvements to computer functionality because the controller's operating code is specifically programmed to cause the controller to distribute tasks to the task pool, as opposed to conventional processing systems in which the controller distributes tasks directly to the co-processors.

172. Even assuming, *arguendo*, that claim 1 is directed to an abstract idea (it is not), claim 1 nonetheless includes inventive concepts involving more than well-understood, routine, and conventional activities previously known to the industry. For example, "the CPU 11 may be configured for use within the system 10 by programming it to recognize and communicate with the task pool 13 and divide the computing requirements

into threads." ('275, 5:54-56). By using the task pool as an intermediary device between the controller and the co-processors, the elements of claim 1, both individually and as a combination, specifically prevent and override the routine and conventional sequence of events performed by prior art processing architectures.

D. Swarm Invented a New Processing Architecture Comprising First and Second Co-Processors, Each Configured to Coordinate Tasks with the Task Pool instead of the CPU.

173. Claim 1 further recites:

> a first co-processor configured to successively: retrieve a first task from the task pool; deliver the first task to the first co-processor; process the first task; generate first resulting data; and update the task pool to reflect completion of the first task, all without any communication between the first co-processor and the controller ('275; 14:28-33); and
>
> a second co-processor configured to successively: retrieve a second task from the task pool; deliver the second task to the second co-processor; process the second task; generate second resulting data; and update the task pool to reflect completion of the second task, all without any communication between the second co-processor and the controller. ('275; 14:34-39).

174. The '275 specification describes the configuration and operation of the first and second co-processors:

> Various embodiments of a parallel processing computing architecture include a CPU configured to populate a task pool, and one or more co-processors configured to proactively retrieve threads (tasks) from the task pool. Each co-processor notifies the task pool upon completion of a task, and pings the task pool until another task becomes available for processing. In this way, the CPU communicates directly with the task pool, and communicates indirectly with the co-processors through the task pool. ('275; 2:19-27);
>
> ...
>
> Upon retrieving a task from the task pool, a cell may then process that task, typically by retrieving data from a particular location in first memory 304, processing that data, and storing the processed data at a particular location within second

> memory 306. When a task is completed, the cell notifies the task pool, the task pool marks the task as completed, and the task pool notifies the CPU that the task is completed. ('275; 11:37-44);
>
> ...
>
> Significantly, the retrieval of tasks and the processing of data by the cells may occur without direct communication between the CPU and the various cells. ('275; 11:47-50).

175. The first and second co-processors, both individually and in combination with each other and/or one or more additional co-processors, improve the operation of a computer by retrieving tasks from a task pool (rather than from the CPU). The co-processors further improve the operation of computers by updating the task pool to reflect task completion, as opposed to conventional processing architectures in which the co-processors directly update the CPU.

176. Even assuming, *arguendo*, that claim 1 is directed to an abstract idea (it is not), claim 1 nonetheless includes numerous inventive concepts. For example, the first and second co-processors are specifically programmed to retrieve respective tasks from the task pool, and subsequently update the task pool after completing their respective tasks, without directly communicating with the controller.

177. Moreover, the specification refers to the co-processors as autonomous, proactive solidarity cells. In this context, the term autonomous implies that a co-processor may interact with the task pool without being instructed to do so by the CPU or by the task pool. The term proactive suggests that each co-processor may be configured (*e.g.*, programmed) to periodically send an agent to monitor the task pool for available tasks appropriate to that co-processor. The term solidarity implies that co-processing cells share a common objective in monitoring and executing all available tasks within the task pool. Prior to Swarm's invention, these inventive concepts had never been proposed before, and thus they involve more than well-understood, routine, and conventional activities previously known to the industry.

E. Swarm Invented a New Processing Architecture Configured to Dynamically Accept the First, Second, and an Additional Co-Processor on a Plug-and-Play Basis.

178. Claim 1 further recites:

> herein the collaborative intelligence system is configured to dynamically accept the first co-processor, the second co-processor, and an additional co-processor into the processing system on a plug-and-play basis without any communication with the controller ('275; 14:40-44).

179. The '275 specification describes the dynamic plug-and-play feature of the invention:

> [I]nteroperability among the CPU and co-processors may be facilitated by configuring the CPU to compose and/or structure tasks at a level of abstraction which is independent of the instruction set architecture associated with the various co-processors, thereby allowing the components to communicate at a task level rather than at an instruction level. As such, devices and their associated co-processors may be added to a network on a 'plug and play' basis. ('275; 3:42-50).

180. Dynamically accepting co-processors on a plug-and-play basis improves the operation of a computer network by integrating co-processors with different instruction set architectures into the same network. ('275; 3:42-52).

181. Even assuming, *arguendo*, that claim 1 is directed to an abstract idea (it is not), claim 1 nonetheless includes numerous inventive concepts. For example, the system may include a plurality of cells, wherein some of the cells are capable of performing the same task types as other cells, to thereby create redundancy in the system. This redundancy allows the system to continue functioning seamlessly when cells are removed from the system or are otherwise unavailable. The system also functions seamlessly when cells are dynamically added to the system. ('275; 6:49-7:2) These inventive concepts had never been proposed before Swarm invented them.

F. Swarm Invented a New Processing Architecture in Which the First and Second Tasks are Associated with a Common Objective.

182. Claim 1 further recites:

> the plurality of first tasks and the plurality of second tasks are associated with a common objective ('275; 14:45-46).

183. The '275 specification describes the relationship of the first and second tasks to a common objective:

> The term solidarity implies that co-processing cells share a common objective in monitoring and executing all available tasks within the task pool. ('275; 2:51-54).

184. Associating the first and second tasks with a common objective improves the operation of a computer network by promoting swarm (or collaborative) intelligence. ('275; 1:1).

185. Even assuming, *arguendo*, that claim 1 is directed to an abstract idea (it is not), claim 1 nonetheless includes numerous inventive concepts. For example, the invention facilitates collaborative intelligence through the use of dynamically configurable proactive autonomous agents. ('275; 1:2-4).

G. <u>Swarm Invented a New Processing Architecture Comprising First and Second Co-Processors Which Autonomously Work Together in Solidarity with the Task Pool to Complete the Common Objective.</u>

186. Claim 1 further recites:

> the first and second co-processors autonomously work together in solidarity with the task pool to complete the common objective ('275; 14:47-49).

187. The '275 specification describes the autonomous action of the co-processors:

> The present invention generally relates to parallel-process computing, and collaborative intelligence, and particularly relates to a processing architecture which involves autonomous co-processors (such as robotic vehicles, Internet of Things (IoT) components, and networked devices) configured to proactively retrieve tasks from a task pool populated by a central processing unit. ('275; 1:17-23).

188. By autonomously working together in solidarity with the task pool to complete the common objective, the first and second co-processors improve the operation

of a computer network by effectively harnessing and exploiting available co-processing resources. ('275; 2:14-15).

189. Even assuming, *arguendo*, that claim 1 is directed to an abstract idea (it is not), claim 1 nonetheless includes numerous inventive concepts. For example, by more effectively harnessing available co-processing resources, the invention reduces CPU management overhead. ('275; 2:13). These inventive concepts had never been proposed before Swarm invented them.

190. Accordingly, claim 1 of the '275 Patent is directed to a new processing architecture which improves the operation of computer, and which includes significantly more than well-understood, routine, and conventional activities.

191. Claims 2 – 17 of the '275 Patent are also directed to various features of a new processing architecture which improve the operation of computer, and which include significantly more than well-understood, routine, and conventional activities.

192. By way of non-limiting example, claim 2 is directed to a collaborative processing system "wherein the first co-processor is configured to perform at least one of depositing a new task into the task pool and modifying an existing task within the task pool."

193. Claim 4 is directed to a collaborative processing system "wherein the first co-processor is configured to determine when it has available processing capacity, and to dispatch the first agent to the task pool in response to the determination."

194. Claim 5 is directed to a collaborative processing system "wherein the controller and the task pool reside on a monolithic integrated circuit (IC) that is not a component of either the first or second co-processors."

195. Claim 6 is directed to a collaborative processing system wherein:

> when the first agent is retrieving the first task from the task pool, the first source address corresponds to an address associated with the first co-processor, the first destination address corresponds to an address associated with the task pool, and the first payload includes a first function which the first co-processor is configured to perform;

> when the first agent is returning from the task pool, the first source address is the task pool's address, the first destination address is the first co-processor's address, and the first payload includes a descriptor of the first task;

196. Claim 8 is directed to a collaborative processing system "wherein the first co-processor is an unmanned autonomous vehicle configured to operate as at least one of a ground vehicle and an aerial vehicle in connection with defense field operations."

197. Claim 10 is directed to a collaborative processing system "wherein neither the controller nor the task pool are incorporated into either the first or second co-processor."

198. Claim 16 is directed to a collaborative processing system "wherein the plurality of autonomous co-processors retrieve the tasks from the task pool via a wireless data connection."

199. Claim 17 is directed to a collaborative processing system wherein "at least one of the plurality of co-processors includes an agent configured to instruct the task pool to select a task of a compatible type and deliver the selected task to the at least co-processors."

200. As explained in detail in the '275 specification, each of the foregoing claims are directed to improvements to the operation of computer, and include significantly more than well-understood, routine, and conventional activities.

THE '777 PATENT

III. SWARM INVENTED NEW AND USEFUL PROCESSES AND MACHINES WHICH IMPROVE THE OPERATION OF A COMPUTER AND WHICH INCLUDE INVENTIVE CONCEPTS.

201. Claim 1 of the '777 Patent is set forth below in its entirety:

> An apparatus for parallel processing of a large computing requirement, the apparatus comprising:
>
> a central processing unit ("CPU");
>
> a task pool in electronic communication with the CPU; and

> a first solidarity cell in electronic communication with the task pool, the first solidarity cell comprising a first agent configured to proactively retrieve, from the task pool, without requiring an instruction from the CPU, a matching task for the solidarity cell to process;
>
> wherein the CPU populates the task pool by dividing the requirement into one or more threads and placing the threads in the task pool, each thread comprising one or more tasks, and the matching task being one of the tasks;
>
> wherein each task comprises a descriptor, the descriptor containing at least:
>
> a function to be executed; and
>
> a memory location of data upon which the function is to be executed;
>
> wherein the first agent is a data frame comprising:
>
> a source address, a destination address and a payload;
>
> wherein the first agent retrieves the matching task by:
>
> being dispatched by the first solidarity cell to the task pool, during which the source address is the first solidarity cell's address, the destination address is the task pool's address, and the payload comprises a list of functions the first solidarity cell is configured to perform;
>
> searching the task pool for a task that is ready to be processed and has a function that the first solidarity cell can perform; and
>
> returning to the first solidarity cell, during which the source address is the task pool's address, the destination address is the first solidarity cell's address, and the payload comprises the descriptor of the matching task.

A. <u>Swarm Invented a New Processing Architecture.</u>

202. The preamble of Claim 1 recites:

> An apparatus for parallel processing of a large computing requirement, the apparatus comprising: ('777; 7:41-42).

203. The '777 specification describes various parallel processing embodiments, for example in the context of co-processors which work in solidarity:

> to complete a large computational requirement by processing threads and subtasks. ('777; 1:53-55).

204. The claimed apparatus involves new and useful machines and processes, and new and useful improvements to machines and processes. Taken together, the CPU, task pool, and solidarity cell provide substantial advantage over prior art processing systems by dividing a large computational requirement into a plurality of threads and placing the threads in the task pool. Claim 1 is thus directed to improvements to computer functionality, as opposed to merely being directed to an abstract idea. Moreover, these improvements are agnostic to the *type* of activities (whether human or non-human) to be processed.

205. Even assuming, *arguendo*, that claim 1 is directed to an abstract idea involving organizing human activities such as a scrum board (it is not), claim 1 nonetheless includes inventive concepts that amount to significantly more than an abstract idea. For example, Claim 1 comprises a task pool in electronic communication with the CPU, and a first solidarity cell in electronic communication with the task pool. Humans are not capable of electronically communicating with microelectronic devices; rather, such electronic communication is the exclusive domain of computers and computer networks.

B. <u>Swarm Invented a New Processing Architecture Comprising a Task Pool Interposed Between the CPU and the Co-Processors.</u>

206. Claim 1 further recites:

> a central processing unit ("CPU"); ('777; 7:41-43).

207. The '777 specification describes the new processing architecture in terms of the interaction among the task pool, the controller (CPU), and the co-processors:

> A method and apparatus for processing information in parallel uses autonomous computer processing units, referred to herein as solidarity cells, to process instructions intended to be executed by a central processing unit ("CPU"). ('777; 1:59-62).
>
> ...

> The CPU divides the information into one or more tasks. A task may include task threads, which each contain one or more subtasks to be performed. The CPU transmits the tasks to a task pool. Each solidarity cell in the system is connected, physically or wirelessly, to the task pool through a switching fabric. The switching fabric facilitates connections for data transfer and arbitration between all system resources. Each solidarity cell is proactive, in that it obtains a task to perform by sending its agent to the task pool when the solidarity cell has no processing to perform. The agent is a software module that searches the task pool for available tasks that match the cell's instruction set architecture. The solidarity cells may execute the task threads sequentially or in parallel, and independently or collaboratively, depending on recipes provided by the CPU. Interdependent tasks within the task pool may be logically combined as needed by the recipe. The task pool notifies the CPU when a task thread is completed. ('777; 2:1-18).

208. The central processing unit improves the operation of a computer by placing the tasks in the task pool, as opposed to sending them directly to the solidarity cells. More particularly, one drawback of conventional multiprocessing frameworks surrounds the architectural requirement that the CPU divide and distribute the threads to the co-processors. Consequently, a significant amount of the CPU's processing time is consumed in managing the co-processing tasks including: distributing the tasks, in sequential order when needed, to co-processors according to their capabilities; waiting for tasks to be completed before distributing result-dependent threads; responding to interrupts from co-processors every time a task is completed; and responding to other messages from co-processors." ('777; 1:31-41).

209. To address these shortcomings, Mr. Íñiguez invented a revolutionary new parallel processing paradigm, including co-processors configured to proactively retrieve new tasks from the task pool without having to communicate directly with (or wait for) the CPU. This alleviates the management workload on the CPU, while keeping the co-processors busy. ('777; 1:43-45).

210. Even assuming, *arguendo*, that claim 1 is directed to an abstract idea (it is not), claim 1 nonetheless includes inventive concepts involving more than well-understood, routine, and conventional activities previously known to the industry. For example, the CPU may be programmed "to recognize and communicate with the task pool 13 and divide the computing requirements into threads … ." ('777; 3:17-19).

C. **<u>Swarm Invented a New Processing Architecture Comprising a Task Pool in Electronic Communication with the CPU.</u>**

211. Claim 1 further recites:

> a task pool in electronic communication with the CPU ('777; 7:44).

212. The '777 specification describes a task pool in electronic communication with the CPU, for example in the context of:

> The CPU 11 may communicate with the task pool 13 directly or through the switching fabric 14. ('777; 2:56-57);
>
> ...
>
> The ability of the CPU 11 to perform the inventive parallel processing methods within the presently described system 10 depends on the CPU's 11 operating system. Specifically, the CPU 11 is a suitable CPU for the system 10 if its operating system may be programmed to recognize and communicate with the task pool 13 and divide computing requirements into threads as described below. ('777; 3:13-19).

213. Claim 1 is directed to improvements to computer functionality in part because the task pool is in electronic communication with the CPU, as opposed to conventional processing systems in which the CPU distributes tasks directly to the co-processors.

214. Even assuming, *arguendo*, that claim 1 is directed to an abstract idea (it is not), claim 1 nonetheless includes inventive concepts involving more than well-understood, routine, and conventional activities previously known to the industry. For example, the CPU's operating system may be specifically programmed to recognize and communicate with the task pool. ('777; 3:13-19). By using the task pool as an intermediary

device between the CPU and the solidarity cell, the claimed invention specifically prevents and overrides the routine and conventional sequence of events performed by prior art processing architectures.

D. <u>Swarm Invented a New Processing Architecture Comprising a Solidarity Cell Configured to Retrieve a Matching Task From the Task Pool Without Requiring an Instruction from the CPU.</u>

215. Claim 1 further recites:

> a first solidarity cell in electronic communication with the task pool, the first solidarity cell comprising a first agent configured to proactively retrieve, from the task pool, without requiring an instruction from the CPU, a matching task for the solidarity cell to process; ('777; 7:45-49).

216. The '777 specification describes the configuration and operation of the first solidarity cell, for example in the context of:

> an apparatus and method for parallel processing in a multiprocessor system using co-processors that proactively seek threads to process. It is a further object that the co-processors be capable of acting autonomously. It is a further object that the co-processors include an agent that searches a task pool to acquire tasks for the co-processors to perform. ('777; 1:47-52).
>
> ...
>
> In particular, the cells 12A ... n do not require an instruction from the CPU 11 to act ('777; 5:21-22).

217. The solidarity cell improves the operation of a computer by proactively retrieving a matching task from the task pool. The solidarity cell further improves the operation of computers by retrieving a matching task from the task pool without requiring an instruction from the CPU.

218. Even assuming, *arguendo*, that claim 1 is directed to an abstract idea (it is not), claim 1 nonetheless includes numerous inventive concepts. For example, the solidarity cell is equipped with a software agent to retrieve a matching task from the task pool, without directly communicating with the CPU.

219. Prior to Swarm's invention, this inventive concept had never been proposed before, and thus claim 1 involves more than well-understood, routine, and conventional steps.

E. Swarm Invented a New Processing Architecture in Which the CPU Populates the Task Pool with a Matching Task.

220. Claim 1 further recites:

> wherein the CPU populates the task pool by dividing the requirement into one or more threads and placing the threads in the task pool, each thread comprising one or more tasks, and the matching task being one of the tasks; (''777; 7:50-53).

221. The '777 specification describes the matching task:

> In another embodiment, the agent 30A searches the task 22 descriptors for an executable instruction that matches one of the instructions that that cell 12A is capable of executing. When a matching task 22 is found, the agent 30A delivers the descriptor of the matching task 22 to the cell 12A, which begins to process the task 22 ('777; 6:25-31).

222. The CPU improves the operation of a computer network by dividing the requirement into one or more threads, and placing the tasks (including the matching task) in the task pool.

223. Even assuming, *arguendo*, that claim 1 is directed to an abstract idea (it is not), claim 1 nonetheless includes numerous inventive concepts. For example, the agent searches the task pool for an executable instruction that matches one of the instructions that that particular solidarity cell is capable of executing. When a matching task 22 is found, the agent delivers the descriptor of the matching task to its solidarity cell. This inventive concept had never been proposed before Swarm's invention.

F. Swarm Invented a New Processing Architecture Comprising a Task Format Which Includes a Descriptor Defining a Function to be Executed and a Location of the Data Upon Which the Function is to be Executed.

224. Claim 1 further recites:

> wherein each task comprises a descriptor, the descriptor containing at least:
>
> a function to be executed; and
>
> a memory location of data upon which the function is to be executed; ('777; 7:54-58).

225. The '777 specification describes the sub-parts of an exemplary task:

> The descriptor may contain one or more of a specific instruction to be executed, a mode of execution, the location of the data to be processed, and the location for placement of the results, if any. ('777; 4:41-44)
>
> ...
>
> In the preferred embodiment, the descriptor is a data structure containing a header and a plurality of reference pointers to memory locations, and the task 22 includes the memory address of the data structure. The header defines the function or instruction to execute. A first pointer references the location of the data to be processed. ('777; 4:50-56).

226. Defining a common task format improves the operation of a computer network by enabling different types of specialized solidarity cells, which may not be directly compatible with each other, to efficiently process large computational requirements within a heterogeneous computing environment. ('777; 3:49-56).

227. Even assuming, *arguendo*, that claim 1 is directed to an abstract idea (it is not), claim 1 nonetheless includes numerous inventive concepts. For example, claim states that a task may include a descriptor comprising: i) a function to be executed; and ii) a memory location of data upon which the function is to be executed ('777; 7:54-58). This inventive concept had never been proposed before Swarm's invention.

G. <u>Swarm Invented a New Processing Architecture Including a Data Frame Comprising a Source Address, a Destination Address, and a Payload.</u>

228. Claim 1 further recites:

> wherein the first agent is a data frame comprising:
>
> a source address, a destination address and a payload; ('777; 7:59-60).

229. The ’777 specification describes an agent in a data frame format:

> An agent 30A, B, C, D ... n, hereinafter collectively referred to as agent 30A ... n to indicate that the system 10 has the same number of agents as solidarity cells 12A ... n, may be considered a data frame in the networking sense. It contains a source address, a destination address, and a payload. (’777; 5:28-33).

230. The claimed agent improves the operation of a computer by selectively retrieving only those tasks which are appropriate for that particular agent.

231. Even assuming, *arguendo*, that claim 1 is directed to an abstract idea (it is not), claim 1 nonetheless includes numerous inventive concepts. For example, a large computing requirement may be parsed into tasks of “a first type, a second type, and a third type; a first cell 12A is capable of performing only tasks of the first type; a second cell 12B can perform tasks of the second type; a third cell 12C can perform tasks of the third type; a fourth cell 12D can perform tasks of the second or third types; and a fifth cell 12N can perform all three task types.” (’777; 3:62-4:2). This inventive concept had never been proposed before Swarm’s invention.

H. <u>Swarm Invented a New Processing Architecture Comprising a Solidarity Cell Configured to Retrieve a Matching Task by Dispatching an Agent to the Task Pool.</u>

232. Claim 1 further recites:

> wherein the first agent retrieves the matching task by:
>
> > being dispatched by the first solidarity cell to the task pool, during which the source address is the first solidarity cell's address, the destination address is the task pool's address, and the payload comprises a list of functions the first solidarity cell is configured to perform; (’777; 7:61-67).

233. The ’777 specification describes an agent being dispatched by its solidarity cell to the task pool:

> To acquire a task 22, a cell 12A sends an agent 30A to the task pool 13 to search for and retrieve an available task 22 that

> requires completion, is not locked, and has a task type that can be performed by the cell 12A. ('777; 5:25-28).
>
> ...
>
> When the agent 30A ... n is dispatched to the task pool 13, the payload contains identifying information of the types of tasks the corresponding cell 12A ... n can perform. When the agent 30A ... n returns from the task pool 13, the payload contains the descriptor of the task 22, either in the form of a memory location or the full descriptor data structure. ('777; 5:55-60).

234. The solidarity cell improves the operation of a computer by dispatching an agent to the task pool, along with a payload which permits the agent to selectively retrieve tasks which are appropriate for that agent.

235. Even assuming, *arguendo*, that claim 1 is directed to an abstract idea (it is not), claim 1 nonetheless includes numerous inventive concepts. For example, claim 1 recites that each solidarity cell's payload includes "a list of functions which that solidarity cell is configured to perform." This inventive concept had never been proposed before Swarm's invention.

I. Swarm's Invented a New Processing Architecture Comprising a Solidarity Cell Configured to Retrieve a Matching Task by Selecting an Appropriate Task From the Task Pool.

236. Claim 1 further recites:

> searching the task pool for a task that is ready to be processed and has a function that the first solidarity cell can perform; and ('777; 8:1-3).

237. The '777 specification describes an agent searching the task pool to retrieve an appropriate task:

> The agent is a software module that searches the task pool for available tasks that match the cell's instruction set architecture. ('777; 2:10-13).

238. The claimed agent improves the operation of a computer by searching the task pool for a task that has a function which the first solidarity cell can perform.

239. Even assuming, *arguendo*, that claim 1 is directed to an abstract idea (it is not), claim 1 nonetheless includes numerous inventive concepts. For example, a large computing requirement may be parsed into tasks of "a first type, a second type, and a third type; a first cell 12A is capable of performing only tasks of the first type; a second cell 12B can perform tasks of the second type; a third cell 12C can perform tasks of the third type; a fourth cell 12D can perform tasks of the second or third types; and a fifth cell 12N can perform all three task types." ('777; 3:62-4:2). This inventive concept had never been proposed before Swarm's invention.

J. <u>Swarm Invented a New Processing Architecture Comprising a Solidarity Cell Configured to Return the Matching Task to the Cell.</u>

240. Claim 1 further recites:

> returning to the first solidarity cell, during which the source address is the task pool's address, the destination address is the first solidarity cell's address, and the payload comprises the descriptor of the matching task. ('777; 8:4-8).

241. The '777 specification describes an agent returning to the solidarity cell from the task pool:

> The source and destination addresses may serve two functions. First, the addresses guide transmission of the agent 30A ... n. The destination address is the address of the task pool 13 when the agent 30A ... n is seeking a task 22, and is the address of the corresponding cell 12A ... n when the agent 30A ... n is returning with a task 22. Correspondingly, the source address is the address of the cell 12A ... n when the agent 30A ... n is seeking a task 22, and is the address of the task pool 13 when the agent 30A ... n is returning with a task 22. ('777; 5:33-42).

242. The solidarity cell improves the operation of a computer by performing the following steps to retrieve a task: i) dispatching an agent to the task pool with a payload identifying the types of tasks the cell can perform; ii) searching the task pool for an appropriate task; and iii) returning to the solidarity cell with a payload identifying the descriptor of the matching task.

243. Even assuming, *arguendo*, that claim 1 is directed to an abstract idea (it is not), claim 1 nonetheless includes numerous inventive concepts. For example, the solidarity cell is configured such that the destination address is the address of the task pool when the agent is seeking a task, and the destination address is the address of its solidarity cell when the agent is returning with a task 22. In contrast, the source address is the address of the solidarity cell when the agent is seeking a task, and the source address is the address of the task pool when the agent returns a task to the cell. These inventive concepts had never been proposed before Swarm invented them.

244. Accordingly, claim 1 of the '777 Patent is directed to a new processing architecture which improves the operation of computer, and which includes significantly more than well-understood, routine, and conventional activities.

245. Claims 2 – 14 of the '777 Patent are also directed to various features of a new processing architecture which improve the operation of computer, and which include significantly more than well-understood, routine, and conventional activities.

246. By way of non-limiting example, claim 2 is directed to an apparatus "wherein the task pool notifies the CPU when the tasks of a thread are completed."

247. Claim 3 is directed to an apparatus "wherein the tasks each comprise a task type selected from a set of task types, and wherein the first solidarity cell is configured to perform tasks of one or more of the task types."

248. Claim 4 is directed to an apparatus "wherein the matching task is a task that is ready to be processed and has a task type that the first solidarity cell can perform."

249. Claim 8 is directed to an apparatus "wherein the descriptor further contains a memory location where processed data is to be stored."

250. Claim 9 is directed to an apparatus "wherein the descriptor is a data structure and the task contains a reference to the memory location of the descriptor."

251. Claim 10 is directed to an apparatus "wherein the task pool occupies a region of physical memory."

252. Claim 11 is directed to an apparatus "wherein the task pool is disposed in a hardware block dedicated to the task pool."

253. As explained in detail in the '777 specification, each of the foregoing claims are directed to improvements to the operation of computer, and include significantly more than well-understood, routine, and conventional activities.

AWS' PRODUCTS AND SERVICES

254. The AWS website describes over 200 cloud-based products and services. Many of these products and services infringe one or more of the Patents-in-Suit either directly under 35 U.S.C. § 271(a), through inducement under § 271(b), and/or by way of contributory infringement under § 271(c).

255. For example, clicking on the Internet of Things product logo on the web page located at https://aws.amazon.com/ reveals a plurality of product families, systems, and sub-systems, including AWS IoT Core and AWS Greengrass. Moreover, the Infringing Products include a feature referred to by Amazon as a "device shadow," and a related feature referred to by Amazon as a "jobs."

256. The attached Claim Charts provide non-limiting illustrations which "map" the Patents-in-Suit to exemplary Infringing Products, and are incorporated herein.

EXEMPLARY CLAIM CHARTS

257. A first claim chart mapping the elements of Claims 1 – 12 of the '004 patent to the Infringing Products, including References 1 – 9 and Figures A - F, is attached hereto as Exhibit W. Exhibit W focuses on the "device shadow" aspect of the Infringing Products; *See* Exhibit Z for a mapping of Claims 1 – 12 of the '004 Patent to the Infringing Products, focusing on the "jobs" aspect.

258. More particularly, with regard to Claim 1 of the '004 Patent, the "processing system" preamble is illustrated, *inter alia*, in Figure A.

259. The "task pool" element may be found at, *inter alia*, Reference 1, page 454; and Reference 4, page 1.

260. The "controller" and "populate" elements may be found at, *inter alia*, Reference 1, page 454; Reference 4, page 1; and Figure B.

261. The "first tasks" and "second tasks" elements generally correspond to the AWS term "shadows" and may be found at, *inter alia*, Reference 1, pages 454 and 494; Reference 7, page 35; Reference 7, page 9; Figure C, D, E, and F; and Reference 7, page 165.

262. The "first co-processor" element generally correspond to the AWS term "device" and may be found at, *inter alia*, Reference 1, page 5; and Reference 5, page 1.

263. The "deliver the first task to the first co-processor" element may be found at, *inter alia*, Reference 1, page 454; and Reference 1, page 218.

264. The "process the first task" element may be found at, *inter alia*, Reference 1, page 5, page 454, and page 460.

265. The "generate first resulting data" element may be found at, *inter alia*, Reference 10, page 1; and Reference 1, page 454.

266. The "and update the task pool to reflect completion of the first task, all without any communication between the first co-processor and the controller" element may be found at, *inter alia*, Reference 1, pages 454 and 494; and Reference 7, page 172.

267. The various elements pertaining to the "second co-processor" which are common to the aforementioned "first co-processor" may be found at, *inter alia*, the same References cited in the above discussion of the "first co-processor."

268. The "wherein the processing system is configured to dynamically accept the first co-processor, the second co-processor, and an additional co-processor into the processing system on a plug-and-play basis without any communication with the controller" element may be found at, *inter alia*, Reference 1, page 494; and Reference 5, page 1.

269. In similar fashion, Exhibit W also maps the elements of claims 2 – 12 of the '004 Patent to the Infringing Products.

270. A second claim chart mapping the elements of Claims 1 – 17 of the '275 patent to the Infringing Products, including References 1 – 9 and Figures A - F, is attached hereto as Exhibit X. Exhibit X focuses on the "device shadow" aspect of the Infringing Products; *See* Exhibit Z for a mapping of Claims 1 – 12 of the '275 Patent to the Infringing Products, focusing on the "jobs" aspect.

271. More particularly, with regard to Claim 1 of the '275 Patent, the "collaborative intelligence system" preamble is illustrated, *inter alia*, in Figure A; Reference 1, page 7; Reference 2, page 1; and Reference 3, page 2.

272. The "task pool" element may be found at, *inter alia*, Reference 1, pages 454 and 494; Reference 4, page 1; Reference 7, pages 35 and 165; Reference 7, page 9; Figures C, D, E, and F.

273. The "controller configured to populate the task pool with a plurality of first tasks and a plurality of second tasks" element may be found at, *inter alia*, Reference 1, pages 454 and 494; Reference 4, page 1; Reference 7, pages 35 and 165; Reference 7, page 9; and Figures B, C, D, E, and F.

274. The "first co-processor" element may be found at, *inter alia*, Reference 1, page 5; and Reference 5, page 1.

275. The "proactively retrieve a first task from the task pool" element may be found at, *inter alia*, Reference 6, page 11; and Reference 1, page 462.

276. The "process the first task" element may be found at, *inter alia*, Reference 1, page 5, page 454, and page 460.

277. The "generate first resulting data" element may be found at, *inter alia*, Reference 10, page 1; and Reference 1, page 454.

278. The "and update the task pool to reflect completion of the first task, all without any communication between the first co-processor and the controller" element may be found at, *inter alia*, Reference 1, pages 454 and 494; and Reference 7, page 172.

279. The various elements pertaining to the "second co-processor" which are common to the analogous elements pertaining to the aforementioned "first co-processor" may be found at, *inter alia*, the same References cited in the above discussion of the "first co-processor."

280. The "wherein the collaborative intelligence system is configured to dynamically accept the first co-processor, the second co-processor, and an additional co-processor into the processing system on a plug-and-play basis without any communication with the controller" element may be found at, *inter alia*, Reference 1, pages 7 and 494; Reference 2, page 1; and Reference 5, page 1.

281. In similar fashion, Exhibit X also maps the elements of claims 2 – 12 of the '004 Patent to the Infringing Products.

282. A third claim chart mapping the elements of Claims 1 – 14 of the ''777 Patent to the Infringing Products, including various internal references and Figures, is attached hereto as Exhibit Z. Exhibit Z includes Claim charts focusing on both the "device shadow" and "jobs" aspects of the Infringing Products.

FIRST CLAIM FOR RELIEF
Infringement of the '004 Patent (35 U.S.C. § 271)

283. Swarm incorporates and realleges paragraphs 1 through 282 of this Complaint

284. Defendants have infringed and continue to infringe claims 1 – 12 of the '004 Patent by making, using, selling, offering to sell, and/or importing infringing products and services into the United States.

285. Defendants' actions as described herein constitute direct, induced, and/or contributory infringement of the '004 Patent in violation of 35 U.S.C § 271(a), (b), and/or (c).

286. Defendants' actions as described herein constitute infringement of the '004 Patent either literally or under the doctrine of equivalents.

287. As a proximate result of Defendants' infringement of the '004 Patent, Plaintiff has been damaged and Defendants have unfairly profited in amounts to be proven at trial.

288. Defendants' infringement of the '004 Patent has been and continues to be willful, entitling Plaintiff to recover treble damages and/or attorney fees pursuant to 35 U.S.C. § 284.

289. Defendants' knowing, intentional, and/or willful actions make this an exceptional case, entitling Plaintiff to an award of reasonable fees pursuant to 35 U.S.C. § 285.

290. Defendant's direct, inducement, and/or contributory infringement of the '004 Patent has caused and will continue to cause Plaintiff irreparable harm unless they are enjoined by this Court.

SECOND CLAIM FOR RELEIF
Infringement of the '275 Patent (35 U.S.C. § 271)

291. Swarm incorporates and realleges paragraphs 1 through 290 of this Complaint.

292. Defendants have infringed and continue to infringe claims 1 – 17 of the '275 Patent by making, using, selling, offering to sell, and/or importing infringing products and services into the United States.

293. Defendants' actions as described herein constitute direct, induced, and/or contributory infringement of the '275 Patent in violation of 35 U.S.C § 271(a), (b), and/or (c).

294. Defendants' actions as described herein constitute infringement of the '275 Patent either literally or under the doctrine of equivalents.

295. As a proximate result of Defendants' infringement of the '275 Patent, Plaintiff has been damaged and Defendants have unfairly profited in amounts to be proven at trial.

296. Defendants' infringement of the '275 Patent has been and continues to be willful, entitling Plaintiff to recover treble damages and/or attorney fees pursuant to 35 U.S.C. § 284.

297. Defendants' knowing, intentional, and/or willful actions make this an exceptional case, entitling Plaintiff to an award of reasonable fees pursuant to 35 U.S.C. § 285.

298. Defendant's direct, inducement, and/or contributory infringement of the '275 Patent has caused and will continue to cause Plaintiff irreparable harm unless they are enjoined by this Court.

THIRD CLAIM FOR RELEIF
Infringement of the '777 Patent (35 U.S.C. § 271)

299. Swarm incorporates and realleges paragraphs 1 through 298 of this Complaint.

300. Defendants have infringed and continue to infringe claims 1 – 14 of the '777 Patent by making, using, selling, offering to sell, and/or importing infringing products and services into the United States.

301. Defendants' actions as described herein constitute direct, induced, and/or contributory infringement of the '777 Patent in violation of 35 U.S.C § 271(a), (b), and/or (c).

302. Defendants' actions as described herein constitute infringement of the '777 Patent either literally or under the doctrine of equivalents.

303. As a proximate result of Defendants' infringement of the '777 Patent, Plaintiff has been damaged and Defendants have unfairly profited in amounts to be proven at trial.

304. Defendants' infringement of the '777 Patent has been and continues to be willful, entitling Plaintiff to recover treble damages and/or attorney fees pursuant to 35 U.S.C. § 284.

305. Defendants' knowing, intentional, and/or willful actions make this an exceptional case, entitling Plaintiff to an award of reasonable fees pursuant to 35 U.S.C. § 285.

306. Defendant's direct, inducement, and/or contributory infringement of the '777 Patent has caused and will continue to cause Plaintiff irreparable harm unless they are enjoined by this Court.

PRAYER FOR RELIEF

WHEREFORE, Plaintiff prays for the following relief against Defendants, jointly and severally:

A. A judgment that Defendants have infringed one or more claims of each of the Patents-in-Suit;

B. An order and judgment temporarily and permanently enjoining Defendants and their officers, directors, agents, servants, employees, affiliates, attorneys, and all others acting in privity or in concert with them, and their parents, subsidiaries, divisions, successors and assigns, from further acts of infringement;

C. A judgment awarding Plaintiff all damages adequate to compensate for Defendants' infringement, but in no event less than a reasonable royalty, including all pre-judgment and post-judgment interest at the maximum rate permitted by law;

D. A judgment awarding Plaintiff all relief (including money damages) contemplated 35 U.S.C. § 154(d);

E. A judgment awarding Plaintiff all damages including treble damages, based on any infringement found to be willful, pursuant to 35 U.S.C. § 284, together with prejudgment interest;

F. A judgment awarding Plaintiff its costs pursuant to 35 U.S.C. § 284;

G. A judgment finding that this case is exceptional and awarding Plaintiff its attorney fees in accordance with 35 U.S.C. § 285; and

H. Any other remedy to which Plaintiff, may be entitled to or the Court deems just and proper.

DEMAND FOR JURY TRIAL

Pursuant to Federal Rule of Civil Procedure 38(b), Plaintiff requests a trial by jury of all aspects properly triable by jury.

DATED this 26th day of August 2022.

BEUS GILBERT McGRODER PLLC

By *<u>/s/Michael K. Kelly</u>*

Leo R. Beus
Michael K. Kelly
Christine N. Jones
Daniel J. Anderson
701 North 44th Street
Phoenix, AZ 85008-6504
(480) 429-3015
Attorneys for Plaintiff

CERTIFICATE OF SERVICE

I hereby certify that on **August 26, 2022**, I electronically transmitted the foregoing document to the Clerk's Office using the ECF System for filing. A copy of the foregoing was electronically provided by the ECF System to all counsel of record.

/s/Nancy Leahy
Nancy Leahy

EXHIBIT LIST

Exhibit	Title
A	U.S. Patent No. 9,852,004
A2	U.S. Patent No. 9,146,777
B	U.S. Patent No. 10,592,275
C	ACC Annual Report dated May 28, 2020
D	October 25, 2019, Washington Business Journal article entitled "AWS is Amazon's income generator, and more from the company's latest earnings"
E	Amazon.com's Corporate Office
F	August 18, 2020, Arizona Republic article entitled "Amazon to add 500 jobs at expanded tech hub in Tempe"
G	Amazon Fulfillment Center
H	August 19, 2020, City of Goodyear in City News article entitled "Amazon Becomes Goodyear's Largest Employer With New Robotics Facility - The 855,000-square-foot facility will add over 1,000 full-time jobs in Goodyear to its existing employee base"
I	March 4, 2021, Amazon's "Jobs" website
J	Amazon Web Services Active Listings
K	March 14, 2019, article entitled "ASU, Amazon Web Services open Smart City Cloud Innovation Center"
L	January 20, 2021, article entitled "ASU's Smart City Cloud Innovation Canter is 'working backwards' to innovate the future"
M	Amazon's Slideshare™ presentation dated April 25, 2018
N	May 27, 2020, Notice of 2020 Annual Shareholders & Proxy Statement"
O	Amazon.com's Annual Report dated December 31, 2020

P	USPTO's Trademark Electronic Search System (TESS) Records Exemplary Registrations
Q	USPTO's Trademark Electronic Search System (TESS) Records Registrations
R	Research Paper entitled "The Octopus as a Model for Artificial Intelligence: A Multi-Agent Robotic Case Study" written by Alfonso Íñiguez
S	November 23, 2015, Letter to Werner Vogels announcing issuance of U.S. Patent 9,146,777
T	February 10, 2016, Email from Alfonso Íñiguez to Amazon
U	August 10, 2016, Email from Alfonso Íñiguez to Amazon Robotics
V	July 16, 2018, Offer Letter to Jeffry Blackburn
W	U.S Patent No. 9,852,004 Claim Charts
X	U.S Patent No. 10,592,275 Claim Charts
Y	Intentionally Left Blank
Z	U.S Patent Nos. 9,852,004, 10,592,275, and 9,146,777 Additional Claim Charts

VERIFICATION OF PLAINTIFFS

Plaintiff Swarm Technology, LLC, by and through its Member Alfonso Iñiguez, hereby declares under penalty of perjury that it has reviewed the foregoing First Amended Complaint, and the factual allegations contained therein are true and correct to the best of its knowledge, memory, information, and belief, and to those matters stated upon information and belief, it believes them to be true.

Executed on: August 26th, 2022.

SWARM TECHNOLOGY, LLC

By: ____________________
Alfonso Iñiguez
Its: Member