This document serves solely as a demonstration of AlphaClaim's technology and does not advocate legal or administrative action against any entity, including entities mentioned herein. This document neither alleges nor implies that any service or product from these entities infringes, either directly or indirectly, upon any patent or other intellectual property rights, including patents mentioned herein. This document neither alleges nor implies the invalidity or unenforceability of any patent, including patents mentioned herein.

This document was created independently by PriceWire, Inc. using AlphaClaim, for illustrative purposes only, without any third-party compensation or under any client contract or direction.

PriceWire, Inc. makes no representations or warranties regarding the completeness or accuracy of the information contained in this document and expressly disclaims any liability related to the use of such information for any purpose. PriceWire, Inc. is not a law firm, and this document does not provide legal advice.

US20240061713A1 Case Study Report

This case study report reviews how AlphaClaim was used to automatically claim chart 923 references in 1 hour to find a high-quality invalidity ground (a well-motivated combination of two references) for claim 1 of US20240061713A1 (application 18/500,070). This ground could serve as part of an office action.

What is AlphaClaim?

AlphaClaim is a "brain" that computes accurate and detailed claim charts for a variety of document types against a given set of claims, adhering to a preponderance of the evidence standard. AlphaClaim has been aligned on 1000s of PTAB IPR institution and final written decisions.

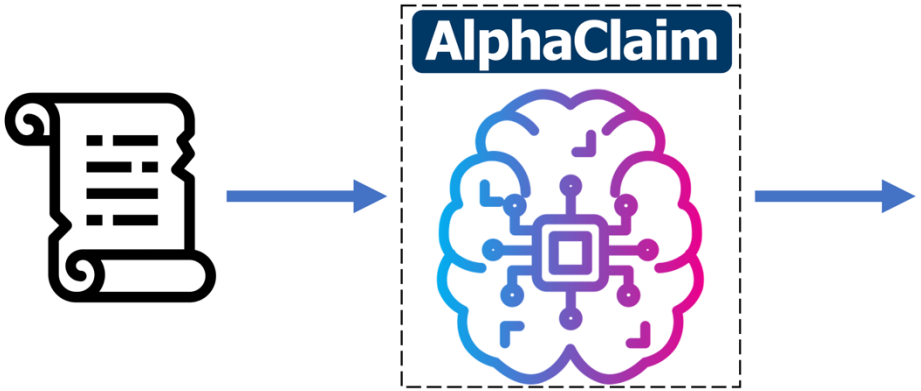
AlphaClaim performs automated, accurate, exhaustive claim charting of **superhuman quantities** of documents (often >10,000). For every document, AlphaClaim produces a claim chart with excerpts and explanations. It then computes an "AlphaScore" out of 5 for each claim element, indicating the strength of the document's disclosure of that element.

AlphaClaim can be applied in three ways. This document focuses on the first.

(1) To **identify the best invalidity grounds** (including combinations), for IPRs or Office Actions.
An AlphaScore of 4 or 5 indicates evidence stronger than the median IPR petition.

(2) To **identify evidence of patent validity** prior to assertion or sale of a patent.
If AlphaClaim's exhaustive review turns up low AlphaScores for all documents, a patent owner can be more confident that the patent will survive IPR or other 102/103 challenges.

(3) To **identify the best evidence of infringement**, for patent owners.
An AlphaScore of 4 or 5 indicates evidence stronger than the median filed claim chart.



Claim Element	Excerpt(s)	Alpha Score	AlphaClaim's Explanation
A method …	"We describe an approach… "	5.0	The excerpt clearly meets the requirements…

How does AlphaClaim work?

AlphaClaim leverages many state-of-the-art AI technologies that are used in systems that achieve quality equivalent to the best humans, as detailed in the chart below. While computationally more expensive than consumer-grade chatbots, AlphaClaim achieves high quality with zero hallucination.

	ChatGPT-4	AlphaCode 2	AlphaGeometry	AlphaClaim
Company	OpenAI	Google	Google	AlphaClaim
Purpose	Chatbot	Coding contests	Math Olympiad	Claim charting
Quality Achieved	Mixed	Top 15% of ranked human competitors	Top ~60 mathematicians, worldwide	Skilled IP attorney
Specialization(s)	None	Fine-tuned, sampling-based	Custom pre-train, symbolic engine	Per-element score, sampling, context extraction
Hallucination Rate	Substantial	Zero	Zero	Zero
Computational Complexity	1x	>1,000x	>1,000x	>1,000x

AlphaClaim has 8 patents pending for its technological innovations.

Case & Invalidity Ground Overview

In this report, we show how in about one hour, AlphaClaim was able to find a high-quality invalidity ground for claim 1 of the '713 patent application.

Patent Application (US20240061713A1, Application 18/500,070)

Filed by Microsoft/Fungible on 11/01/2023. The patent application describes a multi-core DPU (Data Processing Unit) chip for stream processing, including network interfaces and methods to queue up and process packets. The application was published on 02/22/2024. As of May 18, 2024, there has been no office action associated with this application.

How AlphaClaim was used

We used AlphaClaim to automatically, accurately, exhaustively claim chart 923 prior art references. First, we provided AlphaClaim a claim construction, in technical language. This took about 15 minutes. The main AlphaClaim process, claim charting prior art, then took a little under one hour. AlphaClaim charted all the references we provided it, after it removed grace-period prior art. We used AlphaClaim analytics to break up the claim elements and find a combination of references that rendered the claim obvious.

The ground identified

AlphaClaim chose US7,567,567B2 as its base reference with a priority date of 04/05/2005 and a publication date of 07/28/2009. For the first element of the claim ("an integrated circuit comprising"), AlphaClaim relied on the "Cavium Networks OCTEON Plus CN50XX Hardware Reference Manual" v0.99E of July 2008, document file name " CN50XX-HRM-V0.99E", as indexed by Google in 2008.

IDS Tagging

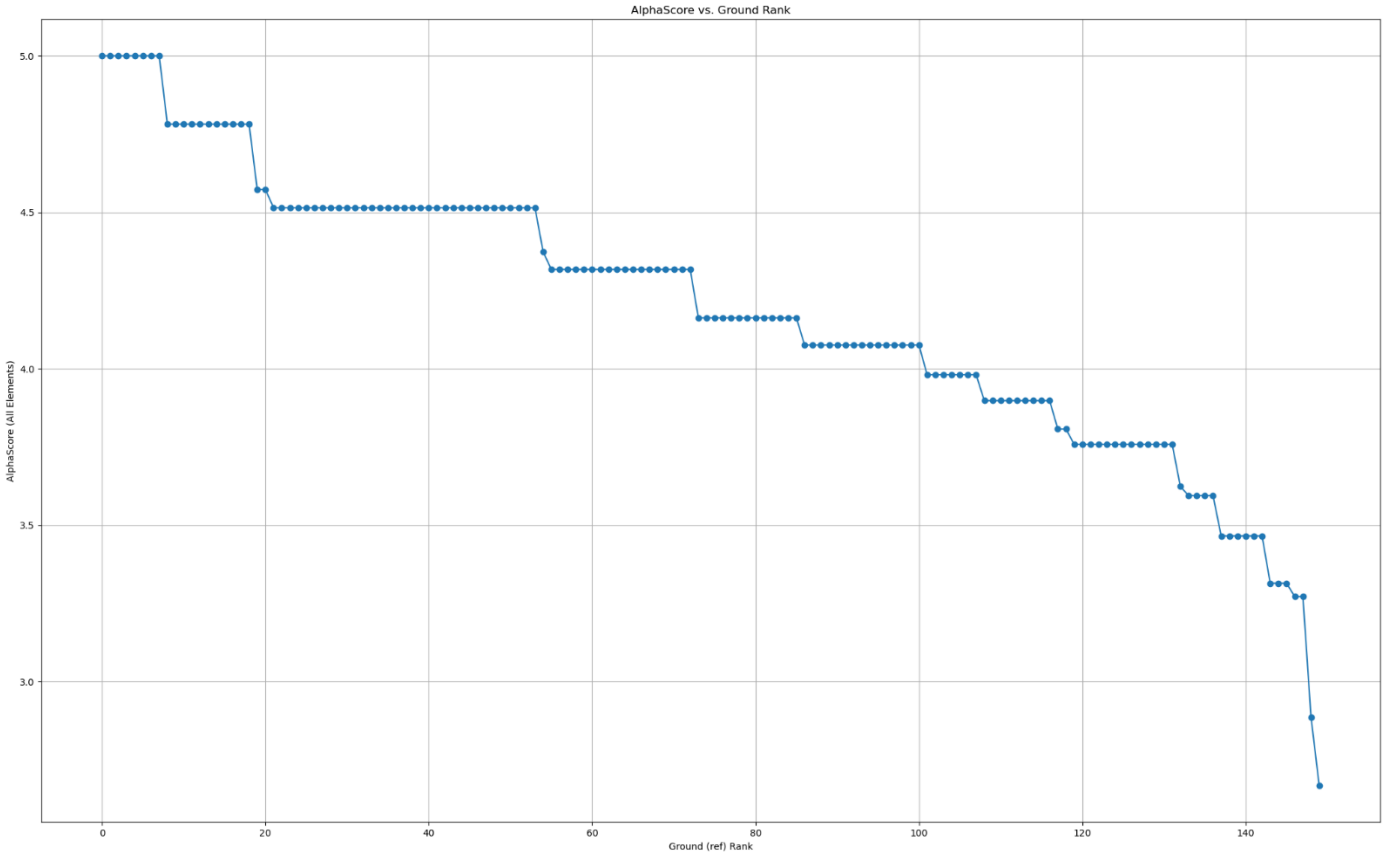
AlphaClaim checks whether the chosen reference, or its family members, are mentioned in the IDS of the target patent or in the IDSes of the family members of the target patent. If any such mention is found, an AlphaIDS tag is generated indicating the type of mention and which specific patent/family are affected. Any references cited by the examiner during prosecution are automatically rejected.

In this instance, no such mention was found for the references cited.

To provide more insight into how AlphaClaim works, we show a score graph for the top 150 references AlphaClaim charted. We then provide the unedited claim chart, AlphaScores, and explanations generated by AlphaClaim for its two-reference ground.

Reviewed Grounds Scores

AlphaClaim works by automatically, accurately, and exhaustively claim charting the prior art provided by the user to find the best invalidity ground possible. The chart shows the results of individual per-reference AlphaScores for the grounds reviewed in this particular case. While AlphaClaim reviewed 923 references for this case, we terminate the chart at 150 documents for viewability, as scores drop too low below that. The per-reference AlphaScore is computed as a mean of the per-element AlphaScores within each reference.



AlphaClaim: Generated Claim Chart

The two chosen references describe similar systems implemented a few years apart, thus enhacing motivation to combine:

Feature	US 7,567,567 B2 Base Reference (2005)	Cavium Networks OCTEON Plus CN50XX Hardware Reference Manual" v0.99E (2008)
at least one interface configured to send and receive stream data units,	In the integrated network interface unit 1300, portions of the four port network interface unit 1200 are included within an integrated solution in which network functions …	Packets and control information can flow to/from CN50XX via any of the RGMII, GMII, MII, or PCI interfaces.
a plurality of programmable processing cores, and	The functional associations of the processing entities 120 a-n are …	High-performance architecture with up to two cnMIPS™ processor cores … CN5020 has two processor cores
a plurality of work unit queues	The network interface unit 110 enables a relatively unconstrained number of queues between a network interface and the processing entities 12	POW implements eight input work queues. The different work queues can be used to provide different service levels.

This is the unedited output from AlphaClaim. Any legal document (e.g., Office Action or IPR petition) would be drafted by a professional based on this output.

Note: the "claim construction" referenced by AlphaClaim is user-provided. This took about 15 minutes to create.

Claim Element (US20240061713A1 claim 1)	Reference: AlphaClaim's Extracted Excerpt	Alpha-Score	AlphaClaim's Explanation
A device comprising: an integrated circuit comprising:	**Cavium Networks OCTEON Plus CN50XX Hardware Reference Manual (2008)**: CN50XX includes many core instructions beyond the standard MIPS64 integer (version 2) instructions. These include instructions to accelerate packet processing, security processing, and memory/cache processing. There is extensive hardware acceleration on CN50XX, but the CPU cores direct the complicated higher-layer application-specific processing. … 1.1.4 Flexible Packet/Control Interfacing Packets and control information can flow to/from CN50XX via any of the RGMII, GMII, MII, or PCI interfaces. The RGMII interface supports up to three ports, and the PCI interface supports one port. This means that internally, CN50XX can support up to a total of four ports. CN50XX hardware efficiently transfers packets via the PCI interface. There is one input and one output port. CN50XX supports all PCI transfer modes and also includes multiple DMA engines, whose features can be used to support per-flow queueing in the PCI host memory.	5.0	1. **Integrated Circuit Type**: The Excerpt describes CN50XX, which appears to be a network processor or communications processor, given its functionalities related to packet processing, security processing, and memory/cache processing. 2. **Packet Processing**: The Excerpt mentions instructions to accelerate packet processing and in-line hardware for packet processing. 3. **Security Processing**: The Excerpt includes instructions for security processing. 4. **Hardware Acceleration**: The Excerpt details extensive hardware acceleration for various tasks. 5. **Core Partitioning**: The Excerpt describes core partitioning for different functions. 6. **Packet/Control Interfacing**: The Excerpt mentions packet and control interfacing via RGMII, GMII, MII, or PCI interfaces. 7. **Work Queuing, Scheduling, Ordering, and Synchronization**: The Excerpt explains work queuing, scheduling, ordering, and synchronization mechanisms, including tag values for ordering/synchronization.

Claim Element (US20240061713A1 claim 1)	Reference: AlphaClaim's Extracted Excerpt	Alpha-Score	AlphaClaim's Explanation
	… In-line Packet-Processing Hardware Acceleration The CN50XX has in-line hardware to offload from the cores all data movement, many common packet-parsing functions, and other important calculations. The in-line hardware completely offloads the work from the cores. The CN50XX in-line packet-processing hardware units complete the following tasks before a core receives a packet: … Hardware Work Queuing, Scheduling, Ordering, and Synchronization CN50XX hardware maintains a work queue in an on-chip hardware unit. This structure provides a primary on-chip communication mechanism between the cores and the hardware units on CN50XX. Both hardware and core software can contribute work to the work queue. The core software can request work at any time, which it becomes aware of either through polling or by interrupts. The hardware schedules the work for the cores. For example, the in-line input packet processing hardware presents an input packet to the CPU cores by creating and submitting a work queue entry to the queue. The core software receives the packet by obtaining the associated work. CN50XX has many ordering/synchronizations mechanisms available to core software, but one important one is closely integrated with the work queueing/scheduling hardware. This hardware orders/synchronizes based on tag values associated with the packet/work.		The Excerpt matches the claim construction very closely. It describes an integrated circuit (CN50XX) that fits the categories of a network processor or communications processor. It covers all required functionalities, including packet processing, security processing, acceleration, core partitioning, packet/control interfacing, packet-processing hardware acceleration, and work queuing, scheduling, ordering, and synchronization. **Note: For this element, we instructed AlphaClaim to look for a network, communications or data processor integrated circuit.**
at least one interface configured to send and receive stream data units,	**Base Reference (2005):** Referring to FIG. 13, a schematic block diagram of an example of an integrated network interface unit 1300 is shown. In the integrated network interface unit 1300, portions of the four port network interface unit 1200 are included within an integrated solution in which network functions are included with a processor core. (The processor core is omitted from the Figure for clarity purposes). More specifically, the integrated network interface unit 1300 includes a transmit DMA module 812, a transmit FIFO reorder logic module 814, a receive FIFO module 816, a receive packet classifier module 818, and a receive DMA module 820. The integrated network interface unit 1200 also includes a media access control (MAC) module 830 and a system interface module 832..	5.0	The excerpt explicitly states the presence of an "integrated network interface unit 1300" which, according to the useful terminology, is an integrated solution that includes portions of a network interface unit with network functions integrated with a processor core. This directly matches the claim construction requirement that the device or processor must have at least one network interface. The excerpt further details the components of the integrated network interface unit, reinforcing the presence of network interface capabilities within the described device. The claim construction is met by the excerpt's description of the integrated network interface unit 1300, which is explicitly stated to include network interface capabilities as part of an integrated solution with a processor core. There is no deviation from the claim construction in the excerpt; it directly states the presence of a network interface within the device, fulfilling the strong requirement set by the claim construction.

Claim Element (US20240061713A1 claim 1)	Reference: AlphaClaim's Extracted Excerpt	Alpha-Score	AlphaClaim's Explanation
a plurality of programmable processing cores, and	**Base Reference (2005):** The functional associations of the processing entities 120 a-n are illustrated by the dashed boundaries designated by reference numerals 310 a-d. The functional association of processing entity 120 a and memory system 130 designated by reference numeral 310 a is a "hypervisor" that is responsible for managing the partitioning and association of the other processing entities, as will be described in greater detail hereinbelow ... In the asymmetrical processing system illustrated in FIG. 3, the specific assignment and mapping of well defined subfunctions or sessions to preassigned processing entities is done to increase efficiency and throughput. Any number of processing entities can be assigned to a processing task that does not directly involve a network interface resource, such as the N processing entities 120c-e. Likewise, any number of processing entities can be assigned to perform a network functionality, protocol or hardware function, such as the M processing entities 120 f-n illustrated in FIG. 3. The present invention uses computer resources for network specific functions that could be low level or high level.	5.0	The Excerpt explicitly mentions the presence of multiple processing entities (120 a-n), which aligns with the requirement from the claim construction for the device to have multiple processors or processing cores. The description of these processing entities being assigned to different tasks, with some handling network functionalities and others handling non-network tasks, further supports the notion of a multi-processor or multi-core system. The mention of a hypervisor managing the partitioning and association of these processing entities implies a complex system designed to utilize multiple processing units efficiently. The Excerpt does not merely imply but explicitly states the involvement of multiple processing entities, which satisfies the claim construction's requirement for the Excerpt to state that the device has multiple processors or processing cores.
a plurality of work unit queues, wherein each work unit queue of the plurality of work unit queues is associated with one of the programmable processing cores and configured to store a plurality of work units enqueued for processing on the one of the programmable processing cores,	**Base Reference (2005):** The network interface unit 110 enables a relatively unconstrained number of queues between a network interface and the processing entities 120. The number of queues is related to the number of processing entities 120. The network interface unit 110 coordinates the servicing of resources such as shared queues. The network interface unit 110 provides for affinity between stateful sessions and processors over time. The network interface unit 110 coordinates interrupt driven packet arrival notification. The network interface unit 110 lowers DMA latency and increases DMA bandwidth to correspond to memory accesses. The network interface unit 110 takes into account multiple packet memory representations. The network interface unit 110 coordinates asynchronous interrupt notifications to minimize processing penalties associated with taking an interrupt. The network system 100 provides multiple queues between a network interface and a plurality of processing entities 120. The multiple queues correspond to the number of processing entities 120. Additionally The network system 100 preserves relevant packet ordering and servicing of multiple queues from multiple processors without coordination overhead..	5.0	1. **One queue per processing core**: The Excerpt explicitly states that the number of queues corresponds to the number of processing entities (120). Given the definition of processing entities as entities that can quickly determine the reason they were activated, this aligns with the claim construction's requirement of having one queue per processing core. 2. **Packet processing queues**: While the Excerpt does not use the phrase "packet processing queues" verbatim, it discusses the coordination of interrupt driven packet arrival notification, lowering DMA latency, and increasing DMA bandwidth, which are all relevant to packet processing. The context of coordinating servicing of resources such as shared queues and preserving relevant packet ordering further implies that the queues are indeed used for packet processing. 3. **Matching Claim Construction**: The Excerpt matches the claim construction by describing a system where the number of queues is directly related to the number of processing cores (entities), and these queues are used in the context of network packet processing. The detailed functionalities described, such as coordination of interrupt notifications, lowering DMA latency, and preserving packet ordering, are all functionalities that support the efficient processing of packets in a system with one queue per processing core.

Claim Element (US20240061713A1 claim 1)	Reference: AlphaClaim's Extracted Excerpt	Alpha-Score	AlphaClaim's Explanation
wherein each work unit of the plurality of work units is associated with one or more of the stream data units; and	**N/A, implied by other elements.**	N/A	N/A
a queue manager executable on a data plane operating system, wherein the queue manager is configured to serialize packet processing such that data packets of a same packet flow are processed by a same programmable processing core.	**Base Reference (2005):** The hash table approach uses a mechanism for load balancing the IP packets to the appropriate processing entity 120 based on transmit affinity. If the operating system 430 wants to drive the affinity from a transmit perspective, then the operating system 430 exploits the hash table interface provided by the network interface unit 110. The application sourcing data running on a particular processing entity 120 (e.g., CPU#n) results in the network interface unit device driver 420 programming the hash table 950 so that received packets for that flow are sent to the Particular processing entity 120 (e.g., CPU#n). The hash table 950 provides the capabilities to manage a large number (e.g., four million) of flows. Each entry in the hash table 950 allows a flow to have a well-defined processing entity 120 plus some pointer, e.g., a pointer to the connection structure ... Before sending out a TCP SYN packet for active open or before sending TCP SYN ACK or TCP ACK, the entries in the hash table 950 are updated according to the processing entity 120 on which the connection is being initiated or terminated as the case may be. Updating the hash table allows subsequent packets for that flow to come to the same processing entity 120.	5.0	The Excerpt precisely matches the claim construction by describing a method (the hash table approach) that maintains packet ordering for packets belonging to the same flow. It ensures that packets of the same flow are processed by the same processing core (processing entity 120) by using a hash table to load balance IP packets based on transmit affinity. The operating system (430) and network interface unit device driver (420) work together to program the hash table (950) to direct packets for a specific flow to a particular processing entity. This is explicitly stated as being done before sending out TCP SYN, TCP SYN ACK, or TCP ACK packets, ensuring that subsequent packets for that flow come to the same processing entity. The Excerpt directly addresses the claim construction's requirements by detailing a method that maintains flow consistency and processor affinity in packet processing.