UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
12/858,912	08/18/2010	James R. Driscoll	07-2105-C2	6876

63710 7590 06/09/2020

INNOVATION DIVISION
CANTOR FITZGERALD, L.P.
110 EAST 59TH STREET (6TH FLOOR)
NEW YORK, NY 10022

EXAMINER
MAGUIRE, LINDSAY M

ART UNIT	PAPER NUMBER
3693	

NOTIFICATION DATE	DELIVERY MODE
06/09/2020	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

lkorovich@cantor.com
patentdocketing@cantor.com
phowe@cantor.com

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

Ex parte JAMES R. DRISCOLL and MATTHEW W. CLAUS

Appeal 2019-001155
Application 12/858,912
Technology Center 3600

Before MICHAEL C. ASTORINO, CYNTHIA L. MURPHY, and TARA L. HUTCHINGS, *Administrative Patent Judges*.

MURPHY, *Administrative Patent Judge*.

DECISION ON APPEAL

The Appellant[1] appeals from the Examiner's rejection of claims 1–7 and 9–23 under 35 U.S.C. § 101 (Rejection I) and 35 U S.C. § 112(b) (Rejection II). We sustain both of these rejections, and, therefore, we AFFIRM.[2]

---

[1] The Appellant is the "applicant" (e.g., "the inventor or all of the joint inventors") as defined in 37 C.F.R. § 1.42. "The real party in interest of the present application is BGC PARTNERS, INC." (Appeal Br. 3.)

[2] We have jurisdiction over this appeal under 35 U.S.C. § 134 and 35 U.S.C. § 6(b).

OVERVEIW OF THE APPELLANT'S INVENTION

According to the Appellant, the claims on appeal are directed to a method/system in which "specific computer operations" allow a computer "to perform a function that improve[s] computer performance." (Appeal Br. 8, emphasis omitted.) In the Appellant's disclosed method/system, the "computer" is a securities-exchange computer "for trading securities." (Spec. 1, l. 5.)

In a typical securities-exchange system, "there is an order book for each security being traded." (Spec. 1, l. 29.) When a trader transmits an "order" to a securities-exchange computer to buy or sell a security, the trader's order is entered into the security's "order book." (*See id.* at 5, ll. 17–18.) Once a trader places an order to buy or sell or security, the trader is legally bound to trade this security at the price specified in his/her order. (*See id.* at 1, ll. 21–23.)

With the Appellant's method/system, a trader transmits a "test order[]" to the securities-exchange computer. (Spec. 5, ll. 17–18.) According to the Appellant, a "test order" is "new kind of order that does not bind the party placing the test order to fulfill a contract to tender the money or securities specified in the test order." (*Id.* at 4, ll. 15–17.) Thus, a "test order" provides a trader with "a means to test trading strategies on a securities exchange" without actually "trading securities." (*Id.* at 4, ll. 8–10.)

When a trader transmits a "test order" to buy or sell a security, the test order cannot be entered into the security's "real order book," because the security is not actually being traded. (*See* Spec. 5, ll. 20–21.) Thus, in the Appellant's method/system, a "test order book" is created. (*See id.* at 5,

ll. 21–24.) More specifically, a test order book is generated as a "data structure." (*See id.* at 3, ll. 21–25, 9, ll. 14–27; 11, ll. 17–31; Figs. 4, 5.)

A securities exchange's overriding economic responsibility resides in the processing of real orders, not test orders. (*See* Spec. 1, ll. 15–23; 4, ll. 15–17.) Thus, it follows that the generation of a data structure for a test order should not cause "substantial detrimental impact on transaction processing." (*Id.* at 12, ll. 4–7.)

INDEPENDENT CLAIMS ON APPEAL
(with our annotations)

1. A method for processing a test order comprising:

[(a)] ***receiving***, by at least one processor, an order for an instrument from a remote computer;

[(b)] automatically ***determining***, by the at least one processor, based on at least one criteria, that the order is to be executed in a test order mode;

[(c)] in response to determining that the order is to be executed in the test order mode, ***identifying***, by the at least one processor, a memory usage of a memory;

[(d)] ***generating***, by the at least one processor, a data structure in a memory for storing the test order, ***if*** the memory usage of the memory is below a threshold, in which the data structure is indicative of an order book;

[(e)] ***ranking***, by the at least one processor, said test order;

[(f)] ***storing***, by the at least one processor, said test order in ***the*** data structure;

[(g)] ***generating***, by the at least one processor, a response for said test order indicating at least whether said test order executed; and

[(h)] ***transmitting***, by the at least one processor, said response for said test order to said remote computer.

> 14. An apparatus comprising a[n] at least one processor and a memory, the memory storing instructions that when executed by the at least one processor direct the at least one processor to:
>
> [(a)] ***receive*** a test order for an instrument from a remote computer;
>
> [(b)] automatically ***determine***, based on at least one criteria, that the order is to be executed in a test order mode;
>
> [(c)] in response to determining that the order is to be executed in the test order mode, ***identify*** a memory usage of the memory;
>
> [(d)] ***generate*** a data structure in the memory for storing the test order, ***if*** the memory usage of the memory is below a threshold, in which the data structure is indicative of an order book;
>
> [(e)] ***rank*** said test order;
>
> [(f)] ***store*** said test order in ***the*** data structure;
>
> [(g)] ***generate*** a response for said test order indicating at least whether said test order executed; and
>
> [(h)] ***transmit*** said response for said test order to said remote computer.

ANALYSIS

Independent claim 1 sets forth a method in which a "processor" performs steps (a)–(h), and independent claim 14 sets forth an apparatus in which a "processor" is programmed to perform steps (a)–(h).[3] (Appeal Br., Claims App.)

Claim Interpretation

Step (a) recites ***receiving*** "an order for an instrument from a remote computer," step (b) recites automatically ***determining*** "that the order is to be executed in a test order mode" (i.e., it is a "test order"), step (c) recites

[3] More particularly, there are "instructions," which, "when executed by" the processor, "direct" the processor to perform steps (a)–(h). (Appeal Br., Claims App.)

identifying "a memory usage" of a "memory," step (d) recites ***generating*** "a data structure in a memory," step (e) recites ***ranking*** "said test order," step (f) recites ***storing*** "said test order in [a] data structure," step (g) recites ***generating*** "a response for said test order indicating at least whether said test order executed," and step (h) recites ***transmitting*** "said response for said test order to said remote computer." (Appeal Br., Claims App.)

Step (d) more particularly recites ***generating*** "a data structure in a memory for storing the test order, ***if*** the memory usage of [a/]the memory is below a threshold, in which the data structure is indicative of an order book." (*Id.*) Thus, step (d) defines a threshold condition that can be met (i.e., memory usage is below the threshold), or, alternatively, not met (i.e., memory usage is not below the threshold).

Step (d) specifies that a data structure is generated ***if*** the threshold condition ***is met***. However, step (d) does not specify what happens if the threshold condition ***is not met***. For example, step (d) does not specify that a data structure is generated ***only if*** the threshold condition ***is met***; or otherwise preclude a data structure from being generated if the threshold condition is ***not met***.

Consequently, independent claims 1 and 14 could read on any of the following three scenarios: 1) the threshold condition ***is met*** and a data structure ***is generated***; 2) the threshold condition ***is not met*** and a data structure ***is not generated***; and 3) the threshold condition ***is not met*** and a data structure ***is generated***.

Rejection I

The Examiner rejects claims 1–7 and 9–23 under 35 U.S.C. § 101 as directed to a judicial exception (a "fundamental economic practice") without

significantly more. (*See* Final Action 3.) More concisely, the Examiner rejects the claims because they do not pass the *Alice* test for patent eligibility.[4] We sustain this rejection.

The 2019 Revised Patent Subject Matter Eligibility Guidance ("2019 § 101 Guidance") provides us with specific steps for discerning whether a claim passes the *Alice* test for patent eligibility. (*See* 2019 Revised Patent Subject Matter Eligibility Guidance, 84 Fed. Reg. 50 (Jan. 7, 2019).) These steps are "[i]n accordance with judicial precedent" and consist of a two-pronged Step 2A and a Step 2B. (*Id.* at 52.)

In the first prong of Step 2A (Prong One), we determine whether the claim "recites" an abstract idea. (2019 § 101 Guidance, 84 Fed. Reg. at 54.) The Guidance "extracts and synthesizes key concepts identified by the courts as abstract ideas," and these concepts include "[c]ertain methods of organizing human activity," and, more particularly "fundamental economic practices." (*Id.* at 52.) For example, a trader "placing an order based on displayed market information is a fundamental economic practice," even when "the claims add a degree of particularity as to how an order is placed." (*Trading Technologies Int'l, Inc. v. IBG LLC*, 921 F.3d 1084, 1092–93 (Fed. Cir. 2019).)

---

[4] In *Alice Corp. v. CLS Bank Int'l*, 573 U.S. 208 (2014), the Supreme Court provided a two-step test to detect when an attempt is being made to patent an abstract idea in isolation. (*Id.* at 217–18.) In first step of the *Alice* test, a determination is made as to whether the claim at issue is "directed to" an abstract idea. (*Id.* at 218.) In the second step of the *Alice* test, a determination is made as to whether "additional elements" in the claim, both individually and as an ordered combination, contribute "significantly more" than the abstract idea. (*Id.* at 217.) In this step, attention is given to whether a claim's additional elements are "conventional." (*Id.*)

The Specification conveys that "receiving orders, storing orders in an order book, matching bids and offers in the order book, executing orders in the order book, and reporting on status of orders and executions" are traditional securities-trading tasks. (Spec. 1, ll. 7–10.) Thus, steps (a), (b), and (e)–(h) recite tasks traditional to securities trading, except that the order is a "test order."

According to the Appellant, a "test order" is "a new kind of order." (Spec. 4, ll. 15–17.) However, such purported newness, at most, introduces a novel financial (i.e., abstract) twist to traditional securities-trading tasks. And "a claim for a *new* abstract idea is still an abstract idea." (*Synopsys, Inc. v. Mentor Graphics Corp.*, 839 F.3d 1138, 1151 (Fed. Cir. 2016).) Even if a test order provides "a trader with additional financial information to facilitate market trades," it is still "an abstract idea." (*Trading Techs. Int'l, Inc. v. IBG LLC*, 921 F.3d 1378, 1384 (Fed. Cir. 2019).)

Consequently, under Prong One of Step 2A, independent claims 1 and 14 recite an abstract idea (i.e., a fundamental economic practice); and we proceed to the second prong of Step 2A (Prong Two).

In Prong Two, we evaluate whether the claim contains additional elements that "integrate" the abstract idea "into a practical application." (*See* 2019 § 101 Guidance, 84 Fed. Reg. at 52.) "[A]dditional elements" are "claim features, limitations, and/or steps that are recited in the claim beyond the identified judicial exception." (*Id.* at 55, n. 24.) Thus, the "additional elements" in independent claims 1 and 14 are those "beyond" the traditional securities-trading tasks discussed above.

Independent claims 1 and 14 require the "processor" to perform, or to be programmed to perform, steps (a), (b), and (e)–(h). (Appeal Br., Claims

App.) The "processor" is an "additional element" beyond the traditional securities-trading tasks recited in steps (a), (b), and (e)–(h). However, claims 1 and 14 simply "instruct the practitioner" to implement these securities-trading tasks (e.g., automatically) with this processor. (*Alice*, 573 U.S. at 225.) This is not enough to "impart patent eligibility." (*Id.* at 223.)

Step (c) and (d) recite limitations involving a "memory," "memory usage," a "threshold" condition ("the memory usage of the memory is below a threshold"), and a "data structure" that is generated ***if*** the threshold condition is met. (Appeal Br, Claims App.) If steps (c) and (d) are beyond the fundamental economic practice of securities trading, they constitute additional elements. And if these steps, individually or in combination, "reflect[] an improvement in the functioning of a computer," this is "indicative" that they "may have integrated the exception into a practical application." (*See* 2019 § 101 Guidance, 84 Fed. Reg. at 55.)

The Appellant argues that "the claims recite specific computer operations that allow the computer to perform a function that improve[s] computer performance." (Appeal Br. 8, emphasis omitted.) The Appellant argues that recited features in the claim "help reduce computer workload including computer resources such as memory resources." (*Id.*) The Appellant argues that "the instant claims recite rules that improve computer performance by ensuring that the data structure does not adversely affect computer performance (e.g., memory usage)." (*Id.* at 12.)

The trouble with the Appellant's position is that the operations, features, and/or rules recited in independent claims 1 and 14 do not correlate with the argued computer improvements.

As noted by the Examiner, "there is nothing recited in the claims that states that the workload on the computer has been reduced." (Answer 4.) This is not fatal, of course, if other claim limitations innovatively lead to a reduction in computer workload. Here, however, the claims do not comport the threshold condition in step (d) with a computer-workload-reducing outcome, nor do the claims specify a ceiling for the value of this threshold. As such, the claimed threshold value could be set so high that the threshold condition is almost always met, and a data structure would be generated even when memory usage is at almost-full capacity. Thus, claims 1 and 14 could allow the generation of a data structure to proceed when its generation would adversely affect computer performance (e.g., memory usage).

Additionally, even if the claimed threshold value did comport with a computer-workload-reducing outcome, the conditional limitation in step (d) renders the recited threshold ineffectual. As discussed above, independent claims 1 and 14 could read on a scenario in which the threshold condition ***is not met*** and a data structure ***is generated***. In this scenario, the data structure would be generated even if memory usage greatly exceeds the threshold. Thus, claims 1 and 14 could still allow the generation of a data structure to proceed when its generation would adversely affect computer performance (e.g., memory usage).

Moreover, even if independent claims 1 and 14 precluded a data structure from being generated when there is insufficient memory resources, this probably would not be enough to integrate a fundamental economic practice into a practical application. If a computer has a certain workload when particular data is processed (e.g., a data structure is generated), it follows that the computer's workload would be reduced when this data is not

processed (e.g., a data structure is not generated). Thus, the Appellant's argued computer improvements appear to be premised solely upon the norm that a computer's workload will differ depending upon whether data is, or is not, processed. This norm would not qualify as "an improvement in the functioning of a computer" worthy of integrating an abstract idea into a practical application. (2019 § 101 Guidance, 84 Fed. Reg. at 55.)

Thus, independent claims 1 and 14 do not integrate the recited fundamental economic practice into a practical application under Prong Two of Step 2A, and so we proceed to Step 2B.

In Step 2B, we evaluate whether the additional elements recited in the claim, individually or in combination, amount to "significantly more" than the abstract idea itself. (2019 § 101 Guidance, 84 Fed. Reg. at 56.) If a claim's additional elements consist conventional computer components, combined to interact in a conventional manner, they will not amount to significantly more than the abstract idea. (*See id.*) In other words, the claim's additional elements do not rescue it under *Alice* step two, and so the claim fails the *Alice* test for patent eligibility.

Here, the Specification defines a "processor" as "one or more microprocessors, central processing units (CPUs), computing devices, microcontrollers, digital signal processors, or like devices or any combination thereof, regardless of the architecture (e.g., chip-level multiprocessing/ multi-core, RISC, CISC, Microprocessor without Interlocked Pipeline Stages, pipelining configuration, simultaneous multithreading)." (Spec. 35, ll. 3–7.) The Specification depicts data structures as "tables" having fields (e.g., columns) which contain bid and offer data. (*See id.* at 9, ll. 14–27, Figs. 4, 5.) The Specification also

describes an electronics exchange network in which the securities-exchange computer and the traders' computers are arranged to interact in a conventional manner. (*See id.* at 7, ll. 1–10, Fig. 1.)

Thus, under Step 2B, the claims' additional elements, when considered individually and as an ordered combination, do not add significantly more to the abstract idea, and so, independent claims 1 and 14 do not pass the *Alice* test for patent eligibility.

Therefore, we sustain the Examiner's rejection of independent claims 1 and 14, and the claims depending therefrom (which are not argued separately), under 35 U.S.C. § 101.

Rejection II

The Examiner rejects claims 1–7 and 9–23 under 35 U.S.C. § 112(b) as being indefinite. (*See* Final Action 2.) We sustain this rejection.

As discussed above, independent claims 1 and 14 could read on a scenario in which the threshold condition ***is not met*** and a data structure ***is not generated*** in step (d). We agree with the Examiner that it is unclear whether steps (e)–(h) are performed, or not performed, with this scenario. Moreover, step (f) expressly requires "storing, by the at least one processor, said test order in ***the*** data structure" (Appeal Br., Claims App., emphasis added), which would not seem to be an option if a data structure was not generated in step (d).

Therefore, we sustain the Examiner's rejection of independent claims 1 and 14, and the claims depending therefrom (which are not argued separately), under 35 U.S.C. § 112(b).

CONCLUSION

Claims Rejected	35 U.S.C. §	Basis	Affirmed	Reversed
1–7, 9–23	§ 101	Eligibility	1–7, 9–23	
1–7, 9–23	§ 112	Indefiniteness	1–7, 9–23	
Overall Outcome			1–7, 9–23	

No time period for taking any subsequent action in connection with this appeal may be extended under 37 C.F.R. § 1.136(a)(1)(iv).

AFFIRMED