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.
14/071,307	11/04/2013	FAUSTO BERNARDINI	END920080067US2	6719

79230 7590 03/02/2018

Law Office of Jim Boice
3839 Bee Cave Road
Suite 201
West Lake Hills, TX 78746

EXAMINER
MEINECKE DIAZ, SUSANNA M

ART UNIT	PAPER NUMBER
3683	

NOTIFICATION DATE	DELIVERY MODE
03/02/2018	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):

Jennifer@BoiceIP.com
Emily@BoiceIP.com
Jim@BoiceIP.com

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

Ex parte FAUSTO BERNARDINI, JARIR K. CHAAR,
YI-MIN CHEE, JOSEPH P. HUCHEL,
THOMAS A. JOBSON JR, DANIEL V. OPPENHEIM, and
KRISHNA C. RATAKONDA

Appeal 2016-008117
Application 14/071,307[1]
Technology Center 3600

Before JAMES R. HUGHES, ERIC S. FRAHM, and
MATTHEW J. McNEILL, *Administrative Patent Judges*.

McNEILL, *Administrative Patent Judge*.

DECISION ON APPEAL

Appellants appeal under 35 U.S.C. § 134(a) from the Examiner's rejection of claims 1–7 and 9–20, which are all the claims pending in this application.[2] We have jurisdiction under 35 U.S.C. § 6(b).

We affirm.

---

[1] According to Appellants, the real party in interest is International Business Machines Corporation. App. Br. 2.

[2] Claim 8 was canceled by amendment on May 20, 2015.

STATEMENT OF THE CASE

Introduction

Appellants' application relates to "the creation of semi-custom software through the use of a standardized software factory." Spec. ¶ 2. Specifically, Appellants describe creating a work packet (which is a "reusable, self-contained, discrete unit of software code that constitute[s] a contractual agreement" Spec. ¶ 33), selecting a human team to complete coding of the work packet, determining whether the human team is competent to create the final work packet, and, if so, ordering the human team to create the final work packet within the software factory. Spec. ¶ 5. Claim 1, reproduced below, is illustrative of the claimed subject matter:

> 1. A computer-implemented method for determining competence levels of factory teams working with a software factory, the method comprising:
>
> configuring, by one or more processors, a software factory, wherein configuring the software factory comprises configuring message routers and communication channels to be used by the software factory, and wherein the software factory comprises:
>
> a software factory governance section that evaluates project proposals for acceptance by the software factory;
>
> a design center composed of a requirements analysis team and an architecture team, wherein the design center sections project proposals into major functional areas that are to be handled by the requirements analysis team and the architecture team, and wherein the design center creates work packets;

an assembly line, wherein the assembly line comprises hardware that executes software that receives and executes work packets to create deliverable software, wherein the assembly line comprises a published set of services and a published set of requirements for the assembly line, wherein the published set of services and the published set of requirements for the assembly line are published to the design center, and wherein the published set of services describes what assembly services for assembling work packets are offered by the assembly line, and wherein the published set of requirements describes what execution environment must be used by work packets that are provided by the design center for assembly in the assembly line, wherein the work packets include governance procedures, standards, reused assets, work packet instructions, integration strategy, schedules, exit criteria and artifact checklist templates for Input/Output routines, wherein the assembly line recognizes project types and automatically assembles work packets needed for a recognized project type, wherein the assembly line conducts an integration test, a system test, a system integration test and a performance test of the deliverable software, wherein the integration test tests the deliverable software for compatibility with the client's system, the system test checks the client's system to ensure that the client's system is operating properly, the system integration test tests for bugs that may arise when the deliverable software is integrated into the client's system, and the performance test tests the deliverable software for defects as it is executing in the client's system;

a software factory analytics and dashboard, wherein the software factory analytics and dashboard monitors a health of the software factory through messages on an Enterprise Service Bus (ESB) that couples endpoint processes of the software factory with dashboard monitors and provides a standard-based integration platform that combines messaging, web services, data transformation and intelligent routing in an event driven Service Oriented Architecture (SOA) via XML data stream messages that

contain factory operation, system, business and performance and activity related metrics; and

a product vendor's interface team, wherein the product vendor's interface team provides an interface between a product vendor and the software factory, wherein the product vendor is an enterprise partner that provides the software factory with supported products that are used by the assembly line to create the deliverable software;

creating, by one or more processors, a template for an initial work packet, wherein the initial work packet is a self-contained work unit that is assembled within the software factory;

creating, by one or more processors, a partially instantiated work packet by populating the template with details that describe pre-conditions and post-conditions necessary to execute the initial work packet, wherein the partially instantiated work packet is not an executable process due to a condition in which roles, associated with activities whose performance is required to execute a work packet, have yet to be assigned to a human team that will work on the work packet, and wherein the pre-conditions comprise software, an operating system, and input data formats required to execute the work packet, and wherein the post-conditions comprise a required output format for displaying an output generated by the work packet;

provisionally selecting, by one or more processors, the human team to perform activities of the partially instantiated work packet;

assigning, by one or more processors, the roles needed to execute the work packet to workers on the human team;

determining, by one or more processors, whether the human team is competent to perform activities of a final work packet; and

> in response to determining that the human team is competent to perform the activities of the final work packet, ordering, by one or more processors, the human team to perform the activities of the final work packet within the software factory.

The Examiner's Rejections

Claims 1–7 and 9–20 stand rejected under 35 U.S.C. § 101 as directed to patent-ineligible subject matter. Final Act. 3.

Claims 1–7 and 9–20 stand rejected under 35 U.S.C. § 112 (pre-AIA), second paragraph, as being indefinite. Final Act. 4–5.

Claims 1–7 and 9–20 stand rejected under the non-statutory doctrine of obviousness-type double patenting as being unpatentable over claims 1–17 of U.S. Patent 8,595,044. Final Act. 6–7.

ANALYSIS

Patent-Ineligible Subject Matter

In *Alice*, the Supreme Court reiterated the two-step framework set forth in *Mayo Collaborative Services v. Prometheus Laboratories, Inc.*, 132 S. Ct. 1289 (2012), for determining whether claimed subject matter is judicially-excepted from patent eligibility under § 101. *Alice Corp. Pty. Ltd. v. CLS Bank Int'l*, 134 S. Ct. 2347, 2355 (2014). Assuming that a claim nominally falls within one of the statutory categories of machine, manufacture, process, or composition of matter, the first step in the analysis is to determine if the claim is directed to a law of nature, a natural phenomenon, or an abstract idea (judicial exceptions). *Alice*, 134 S. Ct. at 2355. For example, abstract ideas include, but are not limited to, fundamental economic practices, methods of organizing human activities, an idea of itself, and mathematical formulas or relationships. *Id.* at 2355–57. If

the claim is directed to a judicial exception, such as an abstract idea, the second step is to determine whether additional elements in the claim "'transform the nature of the claim' into a patent-eligible application." *Id.* at 2355 (quoting *Mayo*, 132 S. Ct. at 1297). This second step is described as "a search for an 'inventive concept'—*i.e.*, an element or combination of elements that is '. . . significantly more than . . . the [ineligible concept] itself.'" *Id.* at 2355 (alteration in original) (quoting *Mayo,* 132 S. Ct. at 1294).

Alice Step One

"[T]he first step in the *Alice* inquiry . . . asks whether the focus of the claims is on the specific asserted improvement in computer capabilities . . . or, instead, on a process that qualifies as an 'abstract idea' for which computers are invoked merely as a tool." *Enfish, LLC v. Microsoft Corp.*, 822 F.3d 1327, 1335–36 (Fed. Cir. 2016). In the present case, we find the focus of claim 1 is on an abstract idea. That is, the method of claim 1 is directed to determining the competence level of a human team to perform certain work, which falls within the category of "'method[s] of organizing human activity'" that the Supreme Court has found to contain abstract ideas. *See Intellectual Ventures I LLC v. Symantec Corp.*, 838 F.3d 1307, 1313 (Fed. Cir. 2016) (alteration in original) (quoting *Alice*, 134 S. Ct. at 2356). Moreover, the language of claim 1 invokes computer technology merely as a tool by reciting that each method step be performed "by one or more processors" without specifically reciting how any processing actually occurs. Accordingly, claim 1, viewed as a whole, does not embody "an improvement to computer functionality itself," *Enfish*, 822 F.3d at 1336, but rather is directed to an abstract idea.

Appellants contend claim 1 is not directed to an abstract idea because there are no cases finding patent-ineligible subject matter that are analogous to the claimed limitations, including "configuring message routers and communication channels" and "an assembly line, wherein the assembly line comprises hardware that executes software." App. Br. 11. Further, Appellants assert "[c]onfiguring message routers and communication channels in a software factory . . . clearly improves the computer-related technology of hardware (i.e., processors) that creates deliverable software." Reply Br. 2–3. We are not persuaded by Appellants' arguments.

We note that we must consider the claim as a whole, not specific limitations in the claim, when considering whether the claim is directed to an abstract idea. *See Internet Patents Corp. v. Active Network, Inc.*, 790 F.3d 1343, 1346 (Fed. Cir. 2015) ("Under step one of *Mayo/Alice*, the claims are considered in their entirety to ascertain whether their character as a whole is directed to excluded subject matter."). Claim 1 as a whole is drawn to a method of determining the competence of a human team to perform certain work, and the limitations of "configuring message routers and communication channels" and "an assembly line . . . comprises hardware that executes software" are merely non-specific uses of computer technology in the service of the abstract idea. Claim 1 does not focus on how any claimed computer technology functions to perform the abstract idea so as to amount to "an improvement to computer functionality itself." *Enfish*, 822 F.3d at 1336. For instance, there is no recitation in claim 1 of how the "message routers and communication channels" are configured so as to provide such an improvement. Further, the mere recitation in claim 1 of computer hardware—e.g., "an assembly line . . . comprises hardware that executes software"—does not in itself save an otherwise abstract idea.

"Software can make non-abstract improvements to computer technology just as hardware improvements can." *Id.* at 1335. Rather, the question is whether the focus of the claim is on an asserted improvement in computer technology. *Id.* 1335–36. For the reasons discussed above, we find that it is not.

Although the guidance in *Enfish* to consider whether a claim focuses on a "specific asserted improvement in computer capabilities," 822 F.3d at 1336, points us toward a conclusion that claim 1 is directed to an abstract idea, we acknowledge that the Supreme Court has not endorsed a "single, succinct, usable definition or test." *Amdocs (Israel) Ltd. v. Openet Telecom, Inc.*, 841 F.3d 1288, 1294 (Fed. Cir. 2016). "Instead of a definition, then, the decisional mechanism courts now apply is to examine earlier cases in which a similar or parallel descriptive nature can be seen—what prior cases were about, and which way they were decided." *Id.* We thus look to similar cases to buttress our findings above, and we disagree with Appellants' contention that claim 1 is not analogous to claims in any other cases found to have been directed to abstract ideas. *See* App. Br. 11.

For example, in *Intellectual Ventures I LLC v. Symantec Corp.*, one of the patents at issue included claims relating to "systems and methods for receiving, screening, and distributing e-mail." *Intellectual Ventures I LLC v. Symantec Corp.*, 838 F.3d 1307, 1316 (Fed. Cir. 2016). The Federal Circuit analogized the claimed invention to a corporate mailroom, where "[s]uch mailrooms receive correspondence, keep business rules defining actions to be taken regarding correspondence based on attributes of the correspondence, apply those business rules to correspondence, and take certain actions based on the application of business rules." *Id.* at 1317.

In this case claim 1 recites "configuring message routers and communication channels to be used by the software factory," "creating . . . a template for an initial work packet," "creating . . . a partially instantiated work packet by populating the template with details that describe pre-conditions and post-conditions necessary to execute the initial work packet," "provisionally selecting . . . the human team to perform activities of the partially instantiated work packet," "assigning . . . the roles needed to execute the work packet to workers on the human team," "determining . . . whether the human team is competent to perform activities of a final work packet," and "ordering . . . the human team to perform the activities of the final work packet within the software factory." In sum, claim 1 relates to a method for establishing a way to communicate in a software factory, defining work to be done, selecting a team of people who are determined to be competent to achieve the work, and ordering the completion of the work by the team. This method is akin to holding a meeting in a corporate boardroom or conference room to manage a team of people to work on a project.

We find, under the first step of *Alice*, that the outcome of claim 1 in this case, which can be analogized to using a corporate boardroom for managing a team of people, should be the same as in *Intellectual Ventures I LLC*, in which the claims were analogous to a corporate mailroom where people manage correspondence. 838 F.3d at 1317. That is, the claims in both cases are directed to abstract ideas related to organizing human activities, in particular, managing business activities. *See also In re Ferguson*, 558 F.3d 1359, 1364 (Fed. Cir. 2009) ("At best it can be said that Applicants' methods are directed to organizing business or legal relationships in the structuring of a sales force (or marketing company).").

We also agree with the Examiner that claim 1 can be properly analogized to other cases where claims were directed to an idea of itself and thus found to be patent-ineligible. *See* Ans. 9–10. In particular, the abstract idea of claim 1—determining the competence of a human team to perform certain work—essentially embodies the idea of making decisions (who to assign work) based on certain facts (the work to be done; the level of competence of certain workers) independent of any particular technology. Our reviewing court has held that claims that embody the idea of decision-making (e.g., financial, diagnostic, organizational) based on certain facts are patent ineligible. *See, e.g., Versata Development Group, Inc. v. SAP America, Inc.*, 793 F.3d 1306, 1333 (Fed. Cir. 2015) ("determining a price, using organizational and product group hierarchies"); *Univ. of Utah Research Foundation v. Ambry Genetics Corp.*, 774 F.3d 755, 763 (Fed. Cir. 2014) ("The methods, directed to identification of alterations of the gene, require merely comparing the patient's gene with the wild-type and identifying any differences that arise.").

Alice Step Two

The second step in the *Alice* analysis requires a search for an "inventive concept" that "must be significantly more than the abstract idea itself, and cannot simply be an instruction to implement or apply the abstract idea on a computer." *Bascom Global Internet Services, Inc. v. AT&T Mobility LLC*, 827 F.3d 1341, 1349 (Fed. Cir. 2016). There must be more than "computer functions [that] are 'well-understood, routine, conventional activit[ies]' previously known to the industry." *Alice*, 134 S. Ct. at 2359 (second alteration in original) (quoting *Mayo*, 132 S. Ct. at 1294).

Appellants contend claim 1 recites significantly more than an abstract idea because "[t]here is no question that the claimed features described

below are not well-known, are not routine, and are not conventional, as evidenced by the fact that the final Office Action deems the presently-pending claims to be allowable subject to the 101/112 rejections" (App. Br. 12), "particularly in view of the fact that there are no 102/103 rejections against any of the pending claims." Reply Br. 3.

We are not persuaded by Appellants' argument. Appellants do not specifically identify which limitations are asserted to add significantly more to the abstract idea. *See* App. Br. 12. Nevertheless, we briefly address certain limitations here. Claim 1 requires each step be performed "by one or more processors." However, this is no more than "an instruction to implement or apply the abstract idea on a computer." *Bascom*, 827 F.3d at 1349. The claim 1 limitation "configuring message routers and communication channels" also does not require any non-conventional computer components. Appellants' Specification does not specifically define "message routers" or "communication channels," but we find the broadest reasonable interpretations of these claim terms encompass generic computers programmed to route messages and generic networking components, respectively.

We also find the claimed "assembly line" does not require more than a generic computer for operation. Indeed, claim 1 broadly recites "wherein the assembly line comprises *hardware that executes software*." Although claim 1 further defines the "assembly line" by listing numerous features, Appellants have not specifically explained why any of the listed features are more than "well-understood, routine, conventional activit[ies]" in the creation of software. *See Alice*, 134 S. Ct. at 2359 (alteration in original) (quoting *Mayo*, 132 S. Ct. at 1294). In any case, claim 1 does not recite how any of the listed features of the "assembly line" bear on the performance of

the claimed method steps. Rather, the "assembly line" is simply part of the "software factory" that provides a context in which the claimed method is performed. In other words, the "assembly line" of the "software factory" is merely an attempt to limit the use of the abstract idea of determining the competence of a human team to perform certain work to the software factory environment. However, "the prohibition against patenting abstract ideas cannot be circumvented by attempting to limit the use of [the idea] to a particular technological environment." *Alice*, 134 S. Ct. at 2358 (alteration in original) (quoting *Bilski v. Kappos*, 130 S .Ct. 3218, 3230 (2010), with internal quotation marks omitted).

Although step two of *Alice* can be satisfied by showing a combination of limitations amounts to significantly more than an abstract idea, *see Bascom*, 827 F.3d at 1347, Appellants have not shown how the combination of recited generic computer components in this case amounts to significantly more than the abstract idea of determining the competence of a human team to perform certain work. Rather, claim 1 simply recites implementing the abstract idea by performing each step in the method by using conventional computer components, e.g., "one or more processors," without any particular non-conventional interaction of recited computer components. *Cf. Bascom*, 827 F.3d at 1350 ("As is the case here, an inventive concept can be found in the non-conventional and non-generic arrangement of known, conventional pieces.").

Lastly, the fact that there are no outstanding anticipation or obviousness rejections is not persuasive of the non-conventionality of the claimed limitations. Although the second step of the *Alice* framework is described as a search for an "inventive concept," 134 S. Ct. at 2355, the analysis is not an evaluation of novelty or non-obviousness. A novel and

non-obvious claim directed to a purely abstract idea is, nonetheless, patent-ineligible. *See Mayo*, 132 S. Ct. at 1304; *see also Synopsys, Inc. v. Mentor Graphics Corp.*, 839 F.3d 1138, 1151 (Fed. Cir. 2016) ("[A] claim for a *new* abstract idea is still an abstract idea. The search for a § 101 inventive concept is thus distinct from demonstrating § 102 novelty.").

We are, therefore, not persuaded the Examiner erred in rejecting as patent-ineligible claim 1, and claims 2–7 and 9–20 not specifically argued separately.

Indefiniteness

The Examiner finds claims 1–7 and 9–20 are indefinite for failing to particularly point out and distinctly claim the subject matter which Appellants regard as the invention. Final Act. 4. Specifically, the Examiner finds, with respect to claim 1, that "[a] software factory is a building or area; therefore, it is confusing to say that the software factory itself is configured, much less that configuring the factory includes configuring message routers and communication channels to be used by the software factory." *Id.* Further, the Examiner finds "[i]t is also not clear how these aspects of the software factory are meant to limit the scope of the method." *Id.*

Appellants contend the claimed "software factory" is not a building, "but rather is a detailed system that includes hardware-based technologies such as message routers, communication channels, etc. that are configured and utilized by the software factory." App. Br. 13.

We agree with Appellants that the Examiner incorrectly interpreted the claimed "software factory" as a building. Appellants' Figure 1 shows an exemplary software factory that includes various governance boards, teams of people, and software processes. *See* Spec. ¶¶ 27–42. The Examiner has not pointed to any description in the Specification to support an

interpretation of a "software factory" as a building. Claim 1 itself recites various aspects of a "software factory," including "a software factory governance section," "a design center," "an assembly line," "a software factory analytics and dashboard," and "a product vendor's interface team," but does not recite a physical building. Accordingly, we disagree with the Examiner's conclusion that the claimed "configuring . . . a software factory" limitation is confusing, and thus indefinite (Final Act. 4), because the Examiner relied on an incorrect interpretation of a "software factory." *See* MPEP § 2173.01(I) ("The first step to examining a claim to determine if the language is definite is to fully understand the subject matter of the invention disclosed in the application and to ascertain the boundaries of that subject matter encompassed by the claim.").

We also disagree with the Examiner that the lack of limitations specifically explaining how the "message routers and communication channels to be used by the software factory" are configured results in a confusing and thus indefinite claim. *See* Final Act. 4. The fact that claim 1 does not recite how the "message routers and communication channels" are configured results in a broad claim, not an indefinite claim. *See In re Miller*, 441 F.2d 689, 693 (CCPA 1971) ("breadth is not to be equated with indefiniteness").

We, therefore, do not sustain the Examiner's indefiniteness rejection of independent claim 1, independent claims 9 and 13 which recite commensurate limitations, and dependent claims 2–7, 10–12, and 14–20.

Double Patenting

Claims 1–7 and 9–20 stand rejected on the ground of non-statutory obviousness-type double patenting over U.S. Patent 8,595,044. Appellants have not argued this rejection is in error. *See* App. Br. 13. Accordingly, we

summarily affirm the non-statutory double patenting rejection of claims 1–7 and 9–20.

DECISION

We affirm the Examiner's decision to reject claims 1–7 and 9–20.

No time period for taking any subsequent action in connection with this appeal may be extended under 37 C.F.R. § 1.136(a). *See* 37 C.F.R. § 41.50(f).

AFFIRMED