



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.
14/668,180	03/25/2015	Matthew Conover	C218	9508
152691	7590	06/22/2020	EXAMINER	
Setter Roche LLP 1860 Blake Street Suite 100 Denver, CO 80202			PATEL, HIREN P	
			ART UNIT	PAPER NUMBER
			2196	
			NOTIFICATION DATE	DELIVERY MODE
			06/22/2020	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):

ipadmin@vmware.com
uspto@setterroche.com

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

Ex parte MATTHEW CONOVER, STEVEN LAWSON, and
JEFFREY ULATOSKI

Appeal 2019-002414
Application 14/668,180
Technology Center 2100

Before JOSEPH L. DIXON, DAVID M. KOHUT, and
JON M. JURGOVAN, *Administrative Patent Judges*.

KOHUT, *Administrative Patent Judge*.

DECISION ON APPEAL

Pursuant to 35 U.S.C. § 134(a), Appellant¹ appeals from the Examiner's final decision to reject claims 1–20, which constitute all claims pending in this application. We have jurisdiction under 35 U.S.C. § 6(b).

We AFFIRM.

¹ We use the word “Appellant” to refer to “applicant” as defined in 37 C.F.R. § 1.42(a). Appellant identifies the real party in interest as VMWARE, INC. Appeal Br. 2.

INVENTION

The present invention is directed to a method and system for “manag[ing] volumes and virtual machines using a location database gathered from a hypervisor management system.” Spec. Abstract. Claim 1 is illustrative of the invention and is reproduced below.

1. A method of operating a volume attachment service to manage volumes and virtual machines, the method comprising:

transferring a location request to a hypervisor management service to identify locations of one or more virtual machines, wherein the one or more virtual machines execute across a plurality of hypervisors managed by the hypervisor management service;

receiving the locations of the one or more virtual machines and storing the locations in a location database, wherein the locations comprise at least a hypervisor identifier for a hypervisor associated with each of the one or more virtual machines; and

in response to a volume action request for a first virtual machine, directing the volume action request to a hypervisor for the first virtual machine based on the locations in the location database.

Appeal Br. 11 (Claims App.)

REFERENCES

Name	Reference	Date
Conover	US 2012/0174096 A1	July 5, 2012
Ota et al.	US 2013/0311989 A1	Nov. 21, 2013
Furusawa et al.	US 2014/0208049 A1	July 24, 2014

REJECTION

Claims 1–20 stand rejected under 35 U.S.C. § 103 as being unpatentable over Conover, Ota, and Furusawa. Final Act. 3–11.

ANALYSIS

Appellant contends Conover does not teach “in response to a volume action request for a first virtual machine, directing the volume action request to a hypervisor for the first virtual machine based on the locations in the location database,” as recited in claim 1. Appeal Br. 7–8; Reply Br. 2–3. In particular, Appellant argues “claim 1 is directed at skipping the communication to the hypervisor management service when the attachment is required, and instead maintains information about the locations of the virtual machine locally at the volume attachment service”; in contrast, “Conover requires the additional step of communicating with the hypervisor management service when the attachment is required rather than relying on locally stored locations of virtual machines to support the attachment.” Reply Br. 2. Appellant further argues “Conover fails to teach that any determination is made at the virtual machine manager [of Conover] to select a hypervisor **from a plurality of hypervisors**,” instead, “Conover merely describes determining what volumes to attach to a virtual machine, and using a hypervisor or datacenter manager to attach the volumes.” Appeal Br. 7–8. Appellant additionally argues “Conover discloses that a local hypervisor may be contacted to attach the volumes, **but never teaches how the hypervisor is selected**,” “fails to teach consulting a [location] database to determine a hypervisor from a plurality of hypervisors,” and “fails to teach that the hypervisor is selected based on locations obtained from the hypervisor management service.” Reply Br. 2–3; Appeal Br. 7–8.

Appellant’s contentions do not persuade us of error in the Examiner’s rejection. At the outset, we note that claim 1 does not require “*skipping* the communication to the hypervisor management service when the attachment

is required” or “*maintain[ing] information about the locations* of the virtual machine *locally at the volume attachment service*,” as Appellant argues. See Reply Br. 2 (emphases added). Claim 1 recites “operating a volume attachment service to manage volumes and virtual machines” using “a location database,” *without specifying where the location database is maintained*. And, claim 1 recites “in response to a volume action request for a first virtual machine, directing the volume action request to a hypervisor for the first virtual machine based on the locations in the location database,” without requiring that *communication with the hypervisor management service be skipped* when directing the volume action request.² Thus, Appellant’s arguments are not commensurate with the scope of claim 1.

Even if (*arguendo*) *skipping* the hypervisor management service were required by claim 1—Appellant’s arguments that Conover does not teach the skipping are unpersuasive. In particular, Appellant’s arguments do not

² We additionally note Appellant’s Specification *does not expressly disallow communication with the hypervisor management service* during volume attachment; instead, the Specification merely provides discussion of non-limiting examples in which communication with the hypervisor management service *may be skipped*. See Spec. ¶¶ 23 (“rather than relying on hypervisor management service 161 to handle every volume action request, volume action service 171 *may distribute* the various volume action requests to the hypervisors, allowing the hypervisors to individually handle the attachment of the volumes”) (emphasis added), 26 (“In some examples, volume action service 171 may inquire hypervisor management service 161 at periodic intervals to maintain an updated database of virtual machine locations” and “[i]n some instances, volume action service 171 *may attempt to use hypervisor management service 161* to process a volume action request. However, if hypervisor management service 161 is overloaded or inoperable, volume action service 171 *may transfer* the volume action request *to the appropriate hypervisor* based on location database”) (emphases added).

address the Examiner’s specific findings that Conover teaches “a virtual datacenter manager [hypervisor management service] responsible for managing several hypervisors” and also “*expressly teaches directing the volume action request to a hypervisor for the virtual machine in response to a volume action request*” when “[t]he request to attach volume is directed to a hypervisor directly by [Conover’s] VM manager.” See Ans. 6–7 (citing Conover ¶¶ 31, 33–34). More particularly, Conover teaches “[o]nce the VM manager 440 has selected the relevant set of storage volumes based on the attach-triggering event, the VM manager 440 *directly* or indirectly *contacts the hypervisor 430* and requests for the storage volumes 412–416 and/or application volumes 419A–419D to be attached.” See Conover ¶ 31 (emphases added). For example, “the VM manager 440 could *directly request* the storage volumes 412–416 and/or application volumes 419A–419D to be attached to the virtual machine 420A *by connecting to the hypervisor 430*,” thereby *skipping the virtual datacenter manager 450* (Examiner’s asserted “hypervisor management service” that is responsible for managing several hypervisors) while directing the volume action request to a hypervisor (Conover’s hypervisor 430). See Conover ¶ 31 (emphases added), Fig. 4. Thus, we disagree with Appellant’s argument that “Conover *requires* the additional step of *communicating with the hypervisor management service* when the attachment is required.” See Reply Br. 2 (emphases added).

With respect to Appellant’s “locations” arguments (e.g., that “Conover fails to teach consulting a [location] database to determine a hypervisor from a plurality of hypervisors” and “fails to teach that the hypervisor is selected based on locations obtained from the hypervisor

management service,” *see* Appeal Br. 8 and Reply Br. 3), we note Appellant’s arguments do not address the Examiner’s combination of teachings from Conover with Ota and Furusawa. Particularly, the Examiner finds Conover suggests using a *virtual machine (VM) identifying location*, as it would be “impossible to communicate the volume attach request for the selected VM to the appropriate hypervisor [(Conover’s hypervisor 430)] without acquiring/known location/identifier/IP address of the hypervisor associated with the selected VM.” Ans. 7 (emphasis omitted). The Examiner further relies on Ota for teaching that VM locations may be stored in a database mapping virtual machines to hosts/hypervisor programs, and relies on Furusawa for teaching that VM locations may include hypervisor identifiers. Final Act. 5–6 (citing Ota ¶¶ 38, 85, Fig. 10; Furusawa ¶ 45, Fig. 6–7, 11, 20–21); Ans. 6, 8–9. We agree with the Examiner’s reasonable findings and conclusions. *See* Ans. 6–9.

Appellant further contends Ota does not teach transferring a location request to *a hypervisor management service* to identify locations of one or more virtual machines, wherein the one or more virtual machines execute across *a plurality of hypervisors managed by the hypervisor management service*, as recited in claim 1 (emphases added). Appeal Br. 8; Reply Br. 3. In particular, Appellant argues Ota’s “configuration module that provides information about a single host and hypervisor is not equivalent or suggestive to a hypervisor management service that **manages a plurality of hypervisors**,” and “Ota fails to teach any configuration where the configuration module may be responsible for *managing a plurality of hypervisors*,” as required by claim 1. Appeal Br. 8 (citing Ota ¶¶ 63, 67, 69, Fig. 5) (emphasis added); *see also* Reply Br. 3.

Appellant’s arguments do not persuade us of Examiner error because they do not address the combination of teachings from Conover, Ota, and Furusawa proposed by the Examiner and including (i) the Examiner’s specific findings that each of Conover and Furusawa teaches the claimed “one or more virtual machines execute across a plurality of hypervisors managed by the hypervisor management service” and (ii) the Examiner’s specific findings that Ota teaches the claimed “transferring a location request to a hypervisor management service to identify locations of one or more virtual machines.” See Final Act. 4–6 (citing Conover ¶¶ 10, 31; Ota ¶¶ 37–38, 69–70, 85, Figs. 6 and 10; Furusawa ¶ 45, Figs. 3, 6–7, 11, and 20–21); Ans. 6–9, 11–12 (citing Conover ¶¶ 33–34; Furusawa ¶ 89); see also Furusawa ¶ 43 (describing Figure 3’s “operations administration network **105**” that is “used for control communications between the administration unit **401** and the hypervisors **207**, and for data transfers regarding the live migrations of the virtual machines **209**”).

Appellant’s arguments also do not address the Examiner’s findings that Ota’s *DRAS* (dynamic resource allocation server) performs “resource management by dynamically allocating resources to the appropriate *VMs*” that execute across a plurality of *hosts/physical servers*. See Final Act. 5 (citing Ota ¶¶ 37–38, 69–70, 85, Figs. 6 and 10) (emphasis added). For example, four *VMs* execute across *host 110A* and four other *VMs* execute across *host 110C* in Ota’s Figure 6, *each host including* “a configuration module 510, and a *Hypervisor program*” whereby “*VMs may be executed on Hypervisor program,*” and Ota’s “*DRAS [implemented on a ‘management*

server of the system’] *manages . . . VMs and Hypervisor programs.*” See Ota ¶¶ 38 (emphases added), 67 (emphases added), 70, Figs. 5–6.³

We are also unpersuaded by Appellant’s argument that the “prior art neither teaches nor suggests *identifying a hypervisor for a virtual machine based on received locations from a hypervisor management service* that manages a plurality of hypervisors.” Appeal Br. 7 (emphasis added). Ota, for example, identifies a hypervisor for a VM based on a table mapping of VMs to hosts/hypervisor programs, the table created by Ota’s DRAS. See Ota ¶ 85, Fig. 10; Final Act. 5 (citing Ota ¶¶ 37–38, 85, Fig. 10).

Appellant further recites additional limitations of claim 1 (“receiving the locations of the one or more virtual machines and storing the locations in a location database, wherein the locations comprise at least a hypervisor identifier for a hypervisor associated with each of the one or more virtual machines”) against alleged deficiencies of Ota. See Reply Br. 3; Appeal Br. 8. Appellant’s discussion of the claimed “receiving” limitations is unpersuasive of Examiner error because Appellant does not address (i) the Examiner’s specific findings that Ota teaches the claimed “receiving the locations of the one or more virtual machines and storing the locations in a location database” and (ii) the combination of teachings from Ota and Furusawa proposed by the Examiner for the claimed “receiving” wherein “the locations comprise at least a hypervisor identifier for a hypervisor associated with each of the one or more virtual machines.” See Final Act. 5–

³ We note that the totality of the teachings of Ota including Ota’s description of virtual machines executing across a plurality of hosts managed by DRAS teaches or suggests that a “hypervisor management service” (Ota’s DRAS) manages a plurality of hypervisors (Ota’s hosts), as claimed. See Ota ¶¶ 37–38, 69–70, 85, Figs. 6 and 10.

6 (citing Ota ¶¶ 37–38, 69–70, 85, Figs. 6 and 10; Furusawa ¶ 45, Figs. 6–7, 11, and 20–21); Ans. 8–9, 11–13.

Therefore, for all the reasons stated *supra*, we sustain the Examiner’s rejection under 35 U.S.C. § 103 of independent claim 1 as unpatentable over the combination of Conover, Ota, and Furusawa. We also sustain the Examiner’s rejection under 35 U.S.C. § 103 of independent claims 10 and 19 for which Appellant provides the same arguments. Appeal Br. 9.

No separate arguments are presented for the dependent claims. *Id.* Thus, for the reasons stated with respect to independent claims 1, 10, and 19, we sustain the Examiner’s rejection of dependent claims 2–9, 11–18, and 20 under 35 U.S.C. § 103 as unpatentable over the combination of Conover, Ota, and Furusawa. *See* 37 C.F.R. § 41.37(c)(1)(iv); *In re King*, 801 F.2d 1324, 1325 (Fed. Cir. 1986); *In re Sernaker*, 702 F.2d 989, 991 (Fed. Cir. 1983).

CONCLUSION

The Examiner’s decision rejecting claims 1–20 under 35 U.S.C. § 103 is affirmed.

DECISION SUMMARY

In summary:

Claims Rejected	35 U.S.C. §	Reference(s)/Basis	Affirmed	Reversed
1–20	103	Conover, Ota, Furusawa	1–20	

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

Appeal 2019-002414
Application 14/668,180

AFFIRMED