UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

NETFLIX, INC.,

Petitioner,

v.

AVAGO TECHNOLOGIES INTERNATIONAL SALES PTE. LIMITED,

Patent Owner.

Inter Partes Review No. IPR2020-01582

U.S. Patent No. 8,572,138 B2

PATENT OWNER'S DEMONSTRATIVES

Pursuant to 37 C.F.R. § 42.70(b) and the Order Setting Oral Argument 37 C.F.R. § 42.70 ("Order") (Paper 23 at 3-4), Patent Owner Avago Technologies International Sales Pte. Limited ("Patent Owner") submits herewith Patent Owner's Demonstratives as required by the Order. Patent Owner also certifies that on December 23, 2021, Patent Owner served Patent Owner's Demonstratives on Petitioner Netflix, Inc., as required by the Order.

Respectfully submitted,

Dated: December 28, 2021

By: /s/Daniel S. Young/

Daniel S. Young, Reg. No. 48,277
Chad E. King, Reg. No. 44,187
ADSERO IP LLC D/B/A/
SWANSON & BRATSCHUN LLC
8210 Southpark Terrace
Littleton, CO 80120
(303) 268-0066 (telephone)
(833) 793-0703 (facsimile)

Counsel for Patent Owner
Avago Technologies International Sales Pte. Limited

CERTIFICATE OF SERVICE

Pursuant to 37 C.F.R. § 42.6, the undersigned hereby certifies that a copy of the foregoing **PATENT OWNER'S DEMONSTRATIVES** was filed through the Patent Trial and Appeal Board End to End (PTAB E2E) electronic filing system on December 28, 2021, with a confirmation copy served via electronic mail, on the following counsel of record for Petitioner:

Harper Batts
SHEPPARD, MULLIN, RICHTER & HAMPTON LLP
email: HBatts@sheppardmullin.com

Jeffrey Liang
SHEPPARD, MULLIN, RICHTER & HAMPTON LLP
email: JLiang@sheppardmullin.com

Chris Ponder
SHEPPARD, MULLIN, RICHTER & HAMPTON LLP
email: CPonder@sheppardmullin.com

SHEPPARD, MULLIN, RICHTER & HAMPTON LLP
email: Legal-Netflix-Broadcom-IPRs@sheppardmullin.com

Dated: December 28, 2021

By: /s/Daniel S. Young/
Daniel S. Young
Registration No. 48,277

United States Patent and Trademark Office

Patent Trial and Appeal Board

NETFLIX, INC.,

(Petitioner)

v.

AVAGO TECHNOLOGIES INTERNATIONAL SALES

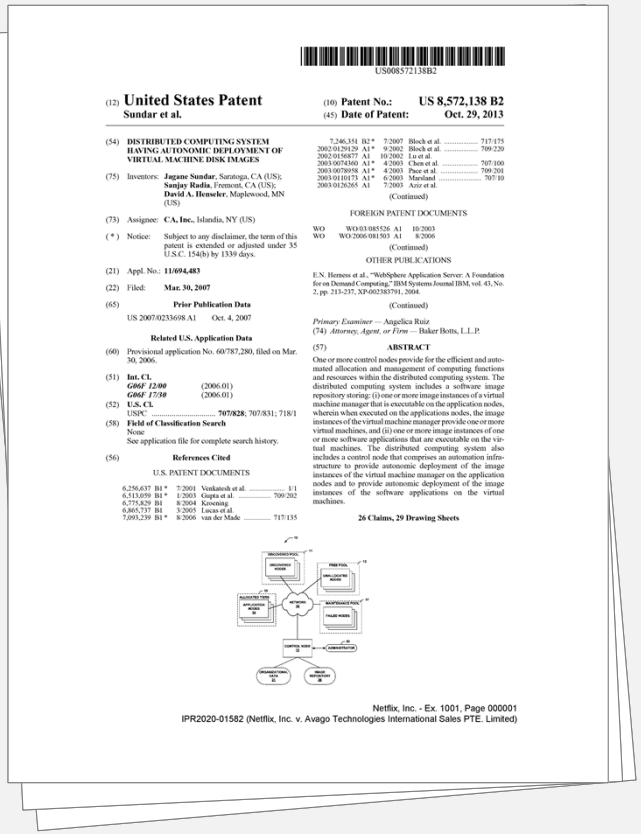
(Patent Owner)

IPR 2020-01582 | U.S. Patent No. 8,572,138

Patent Owner's Demonstrative Exhibits

ADSERO IP

’138 Patent



US008572138B2

(12) **United States Patent**
Sundar et al.

(10) **Patent No.: US 8,572,138 B2**
(45) **Date of Patent: Oct. 29, 2013**

(54) **DISTRIBUTED COMPUTING SYSTEM HAVING AUTONOMIC DEPLOYMENT OF VIRTUAL MACHINE DISK IMAGES**

(75) Inventors: **Jagane Sundar**, Saratoga, CA (US); **Sanjay Radia**, Fremont, CA (US); **David A. Hensler**, Maplewood, MN (US)

(73) Assignee: **CA, Inc.**, Islandia, NY (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 1339 days.

(21) Appl. No.: **11/694,483**

(22) Filed: **Mar. 30, 2007**

(65) **Prior Publication Data**

US 2007/0233698 A1 Oct. 4, 2007

Related U.S. Application Data

(60) Provisional application No. 60/787,280, filed on Mar. 30, 2006.

(51) **Int. Cl.**
G06F 12/00 (2006.01)
G06F 17/30 (2006.01)

(52) **U.S. Cl.**
USPC **707/828**; 707/831; 718/1

(58) **Field of Classification Search**
None
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

6,256,637 B1 * 7/2001 Venkatesh et al. 1/1
6,513,059 B1 * 1/2003 Gupta et al. 709/202
6,775,829 B1 8/2004 Kroening
6,865,737 B1 3/2005 Lucas et al.
7,093,239 B1 * 8/2006 van der Made 717/135
7,246,351 B2 * 7/2007 Bloch et al. 717/175
2002/0129129 A1 * 9/2002 Bloch et al. 709/220
2002/0156877 A1 10/2002 Lu et al.
2003/0074360 A1 * 4/2003 Chen et al. 707/100
2003/0078958 A1 * 4/2003 Pace et al. 709/201
2003/0110173 A1 * 6/2003 Marsland 707/10
2003/0126265 A1 7/2003 Aziz et al.
(Continued)

FOREIGN PATENT DOCUMENTS

WO WO/03/085526 A1 10/2003
WO WO/2006/081503 A1 8/2006
(Continued)

OTHER PUBLICATIONS

E.N. Herness et al., "WebSphere Application Server: A Foundation for on Demand Computing," IBM Systems Journal IBM, vol. 43, No. 2, pp. 213-237, XP-002383791, 2004.
(Continued)

Primary Examiner — Angelica Ruiz
(74) *Attorney, Agent, or Firm* — Baker Botts, L.L.P.

(57) **ABSTRACT**

One or more control nodes provide for the efficient and automated allocation and management of computing functions and resources within the distributed computing system. The distributed computing system includes a software image repository storing: (i) one or more image instances of a virtual machine manager that is executable on the application nodes, wherein when executed on the applications nodes, the image instances of the virtual machine manager provide one or more virtual machines, and (ii) one or more image instances of one or more software applications that are executable on the virtual machines. The distributed computing system also includes a control node that comprises an automation infrastructure to provide autonomic deployment of the image instances of the virtual machine manager on the application nodes and to provide autonomic deployment of the image instances of the software applications on the virtual machines.

26 Claims, 29 Drawing Sheets

Netflix, Inc. - Ex. 1001, Page 000001
IPR2020-01582 (Netflix, Inc. v. Avago Technologies International Sales PTE. Limited)

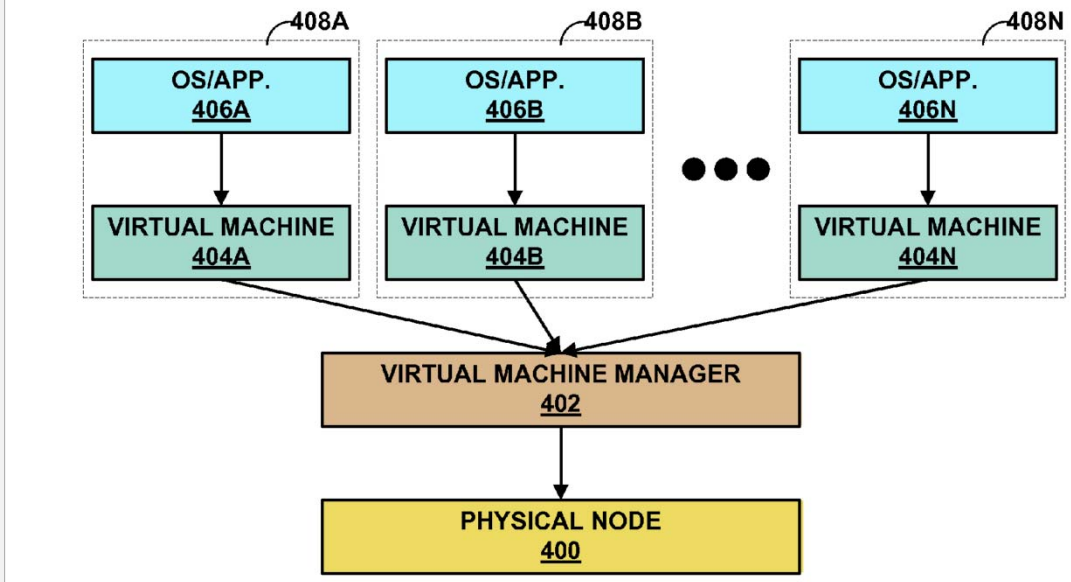


FIG. 24

’138 Patent (Ex. 1001) at Fig. 24; *see* POR at 23-25

’138 Patent | Petitioner’s Invalidity Grounds Fail

GROUND	CLAIMS	PRIOR ART
1	1-5, 9, 19-21, and 26	Khandekar (Ex. 1004)
2	1-5, 9, 19-21, and 26	Khandekar and Le (Ex. 1005)
3	9, 10	Khandekar, Le, and Sidebottom (Ex. 1006)
4	17	Khandekar, Le, and Stone (Ex. 1007)

POR at 10

’138 Patent | Claims Require an Ordered Sequence of Steps

Institution Decision

> [9] The parties are encouraged to address, in the response and reply, whether claim 1 must be construed to require the performance of an ordered sequence of steps, as argued by Patent Owner.

Institution Decision (Paper 9) at 26 n.9; *see* POR at 11

Mformation Techs., Inc. v. Rsch. in Motion Ltd.

> As stated above, a claim requires an ordering of steps when **(1)** the “**specification** directly or implicitly requires an order of steps” **or (2)** “**claim language**, as a matter of logic or grammar, requires that the steps be performed in the order written.”

Mformation Techs., Inc. v. Rsch. in Motion Ltd., 764 F.3d 1392, 1398-400 (Fed. Cir. 2014) (emphasis added)

’138 Patent | Specification Requires an Ordered Sequence of Steps

Specification requires an ordered sequence of steps:

1. all disclosed embodiments describe the creation of image instances of a virtual machine manager and image instances software applications using a “two-phrase capture process”;

2. all disclosed embodiments of the deployment process require a three-step deployment process

POR at 11-31

’138 Patent | Two Sets of Images

’138 Patent

The invention claimed is:

1. A distributed computing system comprising:

a software image repository comprising non-transitory, computer-readable media operable to store: (i) a plurality of image instances of a virtual machine manager that is executable on a plurality of application nodes, wherein when executed on the applications nodes, the image instances of the virtual machine manager provide a plurality of virtual machines, each of the plurality of virtual machine operable to provide an environment that emulates a computer platform, and (ii) a plurality of image instances of a plurality of software applications that are executable on the plurality of virtual machines;

’138 Patent (Ex. 1001) at 36:25-38; *see* POR at 5-6

executing on node **400**. As used herein, a virtual machine manager is a software program that is capable of managing one or more virtual machines. For example, virtual machine

’138 Patent (Ex. 1001) at 32:60-62

manager (e.g., virtual machine manager **402**) (**420**). Administrator **20** may then configure the virtual machine manager to create a virtual machine (e.g., virtual machine **404**) on node **400** (**422**). After creating the virtual machine, administrator

’138 Patent (Ex. 1001) at 35:6-9

tual disk image file. Once configuration is complete, administrator **20** takes a “snapshot” of the data in the virtual disk image file (**430**). Administrator **20** may then capture this “snapshot” to control node **12** as a golden image (**432**). Control node **12** may store the “snapshot” in application matrix **350** as a golden image for use in autonomic deployment of one or more image instances to virtual machines hosting the particular operating system and applications.

’138 Patent (Ex. 1001) at 35:17-24

’138 Patent | Creation of Images Using Two-Phase Capture Process

’138 Patent

A two-phase capture process may be used to provide autonomic control and deployment of the virtual machine disk image files **408** as well as virtual machine manager **402** to physical nodes within distributed computing system **10**. In

’138 Patent (Ex. 1001) at 33:42-45; *see* POR at 12-22

Petitioner’s Expert: Dr. Shenoy

Q. Okay. And the Sundar patent refers to this process as a two-phase capture process; is that correct? And I can highlight for you where I'm referring to. I'm referring to specifically column 33, line 42, which I've highlighted.

A. Yes. In this embodiment that we are discussing, it does refer to a two-phase capture process for capturing the virtual machine manager and the applications.

Testimony of Dr. Shenoy (Ex. 2002) at 40:22-41:8

Phase 1 – Process to Capture Image Instance of Virtual Machine Manager – Figure 24

'138 Patent

A two-phase capture process may be used to provide autonomic control and deployment of the virtual machine disk image files **408** as well as virtual machine manager **402** to physical nodes within distributed computing system **10**. In particular, control node **12** may first capture an image of virtual machine manager **402** installed on physical node **400** prior to installation and configuration of any guest operating system and applications **406**. Control node **12** may utilize this captured image of virtual machine manager **402** as a "golden image" from which instance images can be created for this type virtual machine. The instance images may be stored within application matrix **350** for autonomic deployment as other software images (FIG. **21**).

'138 Patent (Ex. 1001) at 33:42-54; *see* POR at 12-14

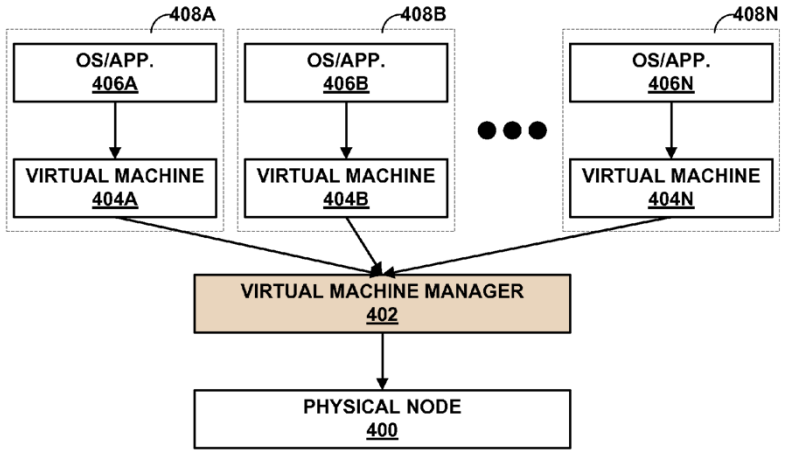


FIG. 24

'138 Patent (Ex. 1001) at Fig. 24

’138 Patent | Two-Phase Capture Process

Phase 1 – Process to Capture Image Instance of Virtual Machine Manager – Figure 24

’138 Patent

A two-phase capture process may be used to provide autonomic control and deployment of the virtual machine disk image files **408** as well as virtual machine manager **402** to physical nodes within distributed computing system **10**. In particular, control node **12** may first capture an image of virtual machine manager **402** installed on physical node **400** prior to installation and configuration of any guest operating system and applications **406**. Control node **12** may utilize this captured image of virtual machine manager **402** as a “golden image” from which instance images can be created for this type virtual machine. The instance images may be stored within application matrix **350** for autonomic deployment as other software images (FIG. **21**).

’138 Patent (Ex. 1001) at 33:42-54; *see* POR at 12-14

Petitioner’s Expert: Dr. Shenoy

Q. Okay. And it states that the first capture is of the image instance of the virtual machine manager; correct?

A. So at line 46, it does talk about control node 12 would capture -- would first capture an image of the virtual machine manager installed on physical node 400. So in this particular example, it talks about doing that first.

Q. And that capture is done prior to the installation and configuration of any guest operating system and applications; correct?

A. So, again, that sentence goes on and says it's prior to the installation and configuration of any guest operating system and applications 406. So in this particular example, yes.

Testimony of Dr. Shenoy (Ex. 2002) at 41:9-42:1

’138 Patent | Two-Phase Capture Process

Phase 1 – Process to Capture Image Instance of Virtual Machine Manager – Figure 25

’138 Patent

FIG. **25** is a flowchart illustrating an exemplary procedure for creating and capturing a golden image for a particular virtual machine manager **402** (FIG. **24**). Initially, administra-

’138 Patent (Ex. 1001) at 34:48-50

Petitioner’s Expert: Dr. Shenoy

A. Okay. Figure 25 is describing a specific embodiment where you first set aside or choose a node for imaging as an image host, that's step 410. Then you install a virtual machine manager on that host, which is 4 -- step 412. You can then configure the virtual machine manager with a desired number of VMs that you want to run on it. That's the next step. You can perform additional configuration of networking and storage area network, SAN, configuration. That's step 416. And in the final step, you capture using the control node -- or you specify, rather, the capture of the virtual machine manager that is being installed on that host.

Testimony of Dr. Shenoy (Ex. 2002) at 47:9-22; *see* POR at 14-16

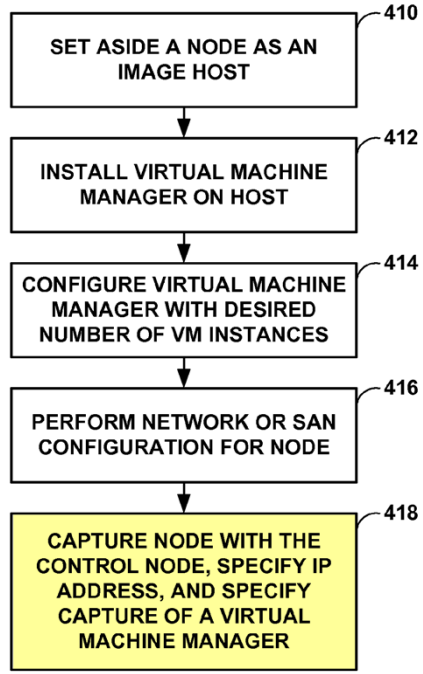


FIG. 25

’138 Patent (Ex. 1001) at Fig. 25

’138 Patent | Two-Phase Capture Process

Phase 2 – Process to Capture Image Instance of Software Applications – Figure 24

’138 Patent

After the first capture is complete, an administrator may install one or more virtual machines **404** to virtual machine manager **402**. After the user installs and configures guest operating systems and any necessary applications **406** on the virtual machines **404**, control node **12** may capture software images for storage as virtual disk image files **408**. Specifically, control node **12** may utilize the captured images as a golden images for the particular operating systems, applications and specific configuration of the virtual machine. The golden image may then be used to produce image instances, i.e., virtual disk images in this case, that are bootable on the virtual machines. The image instances may likewise be stored in application matrix **350** for autonomic deployment.

’138 Patent (Ex. 1001) at 33:55-67; *see* POR at 17

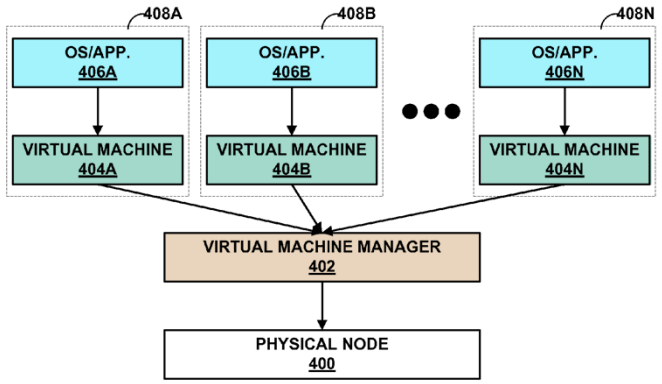


FIG. 24

’138 Patent (Ex. 1001) at Fig. 24

Phase 2 – Process to Capture Image Instance of Software Applications – Figure 26

'138 Patent

FIG. **26** is a flowchart illustrating an exemplary procedure for creating and capturing a virtual disk image file, i.e., a golden image of a guest operating system and applications for execution on a virtual machine. Initially, administrator **20** sets

'138 Patent (Ex. 1001) at 35:1-4; *see* POR at 17-20

Petitioner's Expert: Dr. Shenoy

Q. Okay. So I'm going to go, then, to figure 26. I have figure 26 on the screen. If you could, then, as you did with figure 27, kind of walk us through your understanding of what this flow chart describes?

A. Okay. So this is an example procedure for capturing image instance of software applications. So it starts off by choosing or setting aside a node that is running a virtual machine manager. And it creates a virtual machine on that node. That's the second step. It then installs an operating system and applications and performs any configurations for the OS and applications as needed. Then it takes a snapshot of the virtual disk, of the virtual -- of the VM, and it captures the virtual disk image onto the control node

Testimony of Dr. Shenoy (Ex. 2002) at 57:16-58:9

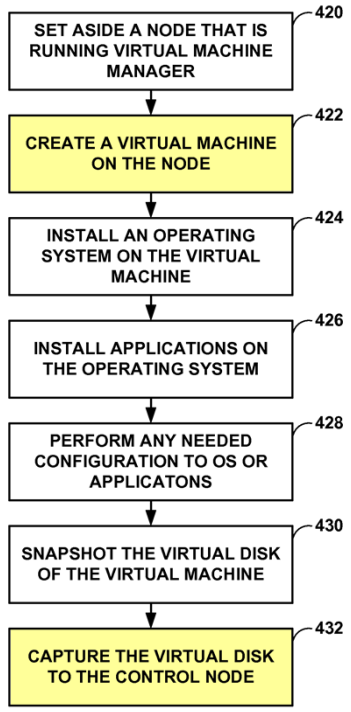


FIG. 26

'138 Patent (Ex. 1001) at Fig. 26

Phase 2 – Virtual Machine Manager Creates the Virtual Machine – Figure 26

'138 Patent

manager (e.g., virtual machine manager **402**) (**420**). Administrator **20** may then configure the virtual machine manager to create a virtual machine (e.g., virtual machine **404**) on node **400** (**422**). After creating the virtual machine, administrator **20** installs the particular operating system on virtual machine **404** (**424**). Administrator **20** may then install one or more applications on the operating system (**426**). After installation,

'138 Patent (Ex. 1001) at 35:6-12

Petitioner's Expert: Dr. Shenoy

Q. And that's the same for this figure, right? If we look at, as an example, look at column 35, lines 6 through 10, in that section, it states that the virtual machine manager creates the virtual machine. For example, virtual machine 404 on node 400; correct?

A. So just to be clear, the sentence that appears in column 35 is referring to figure 26. The figure on the screen is figure 24. So the figure 26 is a flow chart, and it does say that virtual machine manager 402 can create a virtual machine.

Testimony of Dr. Shenoy (Ex. 2002) at 30:17-31:6; *see* POR at 20-22

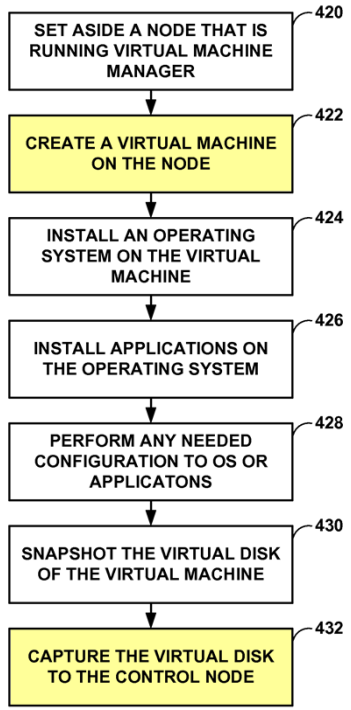


FIG. 26

'138 Patent (Ex. 1001) at Fig. 26

’138 Patent | Three-Step Deployment Process – Figure 24

’138 Patent

402 to one or more nodes in free pool **13**. Once control node **12** deploys the image instance of virtual machine manager **402** to a node, the node appears to be one or more new, unallocated nodes. For instance, if the image instance of virtual machine manager **402** is configured to support five virtual machines, control node **12** represents the newly deployed image instance as though five new, unallocated nodes have been added to free pool **13**. Subsequently, control node **12** may deploy an image instance of virtual disk image files **408** to one of the new unallocated nodes.

STEP 1

STEP 2

STEP 3

’138 Patent (Ex. 1001) at 34:5-13; *see* POR at 22-25

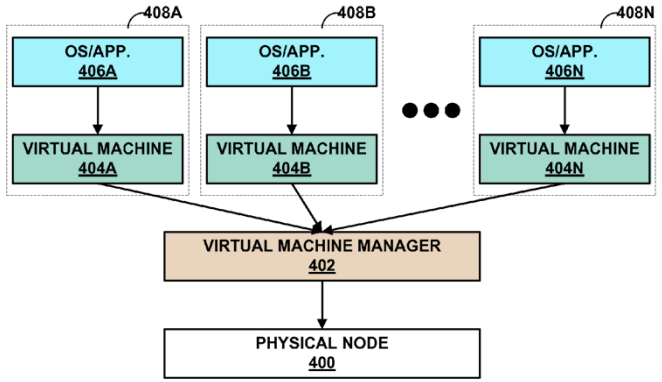


FIG. 24

’138 Patent (Ex. 1001) at Fig. 24

’138 Patent | Three-Step Deployment Process – Figure 24

Petitioner’s Expert: Dr. Shenoy

Q. And as used in Sundar, is a virtual machine manager a software program that is capable of managing one or more virtual machines?

* * *

A. So in the Sundar patent, the virtual machine manager is a software program that manages virtual machines.

Q. Okay. So I think, based on our -- how we arrived here so far, figure 24, which is up on your screen, in part, depicts a virtual machine manager that's executing on an application node; correct?

A. So in figure 24 of -- the physical node is an application node on a physical computing device. The virtual machine manager is executing on that computer or physical node.

Testimony of Dr. Shenoy (Ex. 2002) at 17:3-5; 17:13-18:2; *see* POR at 22-25

’138 Patent

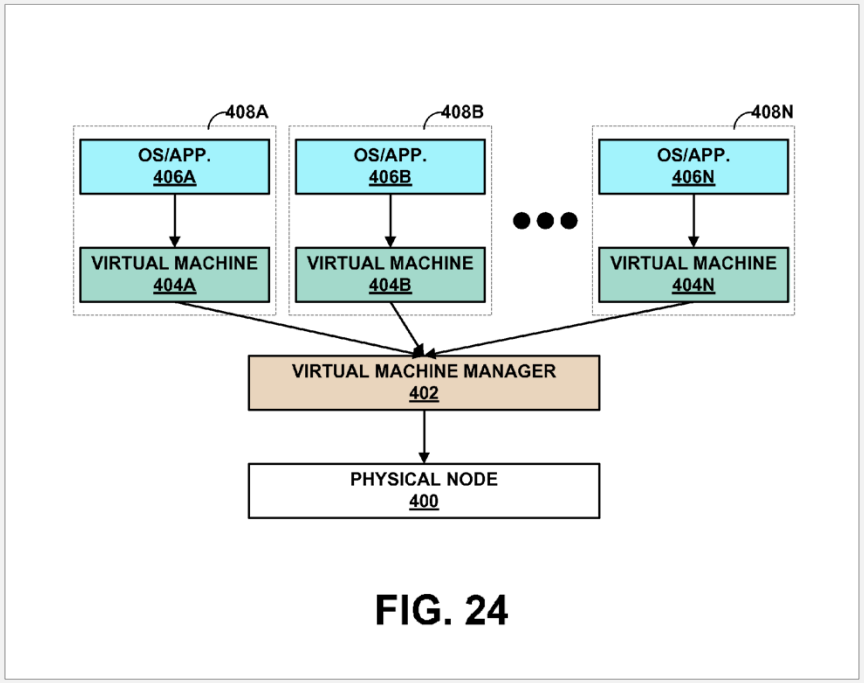


FIG. 24

’138 Patent (Ex. 1001) at Fig. 24

’138 Patent | Three-Step Deployment Process – Figure 24

Petitioner’s Expert: Dr. Shenoy

Q. Okay. And, again, in the Sundar patent, a virtual machine refers to software that creates an environment that emulates a computer platform; correct? And I'm referring to Sundar 32, lines 68 through 33, line 6.

A. As stated in the Sundar patent, starting at column 33, line 1, it says “A virtual machine is used herein to refer to a software that creates an environment that emulates a computing platform.” So virtual machine in the context of the patent, and this example is software that creates an environment to emulate a computing platform.

Testimony of Dr. Shenoy (Ex. 2002) at 18:8-20; *see* POR at 22-25

’138 Patent

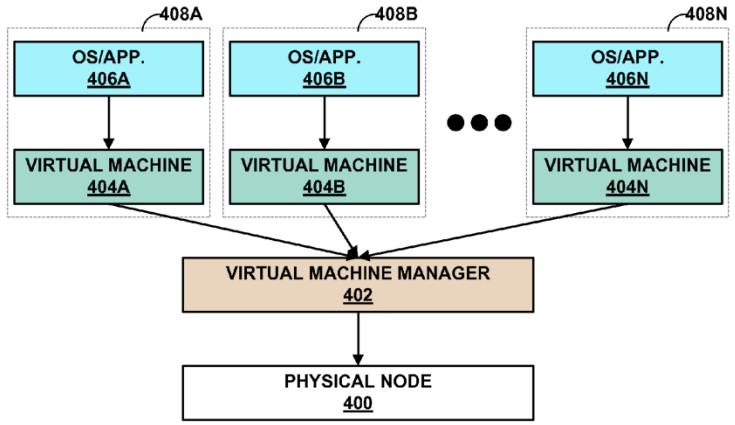


FIG. 24

’138 Patent (Ex. 1001) at Fig. 24

’138 Patent | Three-Step Deployment Process – Figures 27A-B

’138 Patent

FIG. **27**A is a flowchart illustrating an exemplary procedure for autonomic deploying a virtual machine manager to a node. Initially, administrator **20** may instruct control node **12** to create a new tier (**440**). Subsequently, administrator **20** specifies one or more captured virtual machine manager image instances for assignment to the slots of the new tier (**442**). Administrator **20** may also specify whether the image instances of the tier run in persistent or non-persistent mode. Administrator **20** may then activate the new tier (**444**). After these physical nodes boot with the virtual machine manager image instance, each of the nodes creates a set of virtual machines as specified by the image instances. Control node **12** may then discover each of these virtual machines and place them in free pool **13** as if they were independent nodes. In this manner, each of the virtual machines appear to be physical nodes available for use as application nodes **14** when needed. However, control node **12** maintains tags or other metadata that distinguish between virtual machines and actual physical nodes. Finally, based on the current state and goals of distributed computing system **10**, control node **12** may allocate one or more of the virtual machines to the slots within the tier, similar to the process by which nodes are deployed from the free pool **13** to a tier as needed.

STEP 1

STEP 2

’138 Patent (Ex. 1001) at 35:25-47; *see* POR at 26-28

’138 Patent

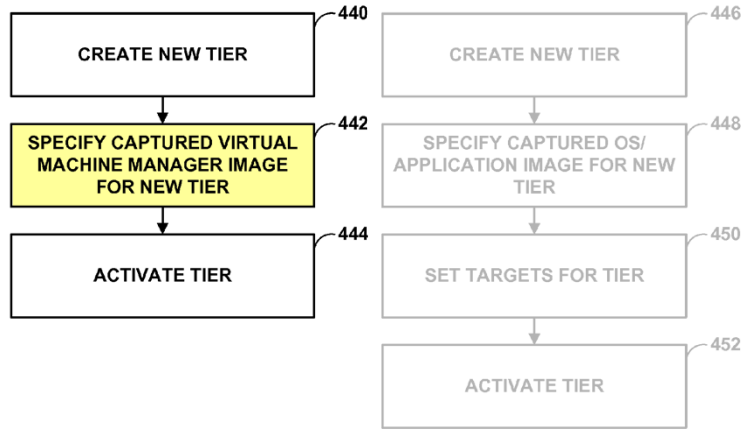


FIG. 27A FIG. 27B

’138 Patent (Ex. 1001) at Fig. 27a-27b

’138 Patent | Three-Step Deployment Process – Figures 27A-B

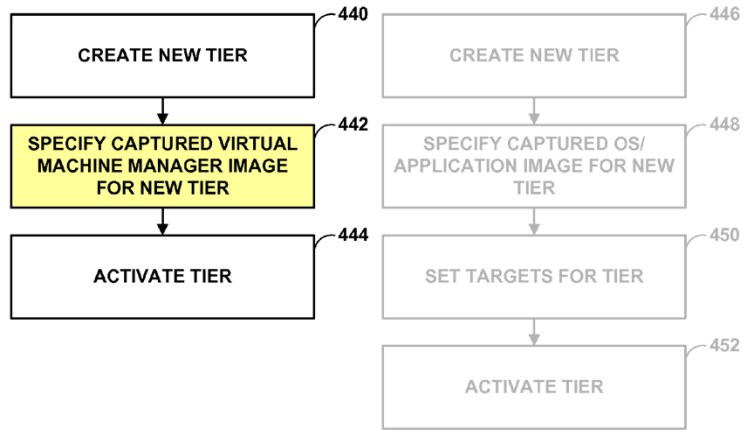
Petitioner’s Expert: Dr. Shenoy

Q. Okay. So in this process, once the image instance of the virtual machine manager is deployed to a node in the free pool and the virtual machine manager is used to create a virtual machine, then that physical machine then becomes available for use as an application node 14; correct?

A. So in this case, once the virtual machine manager has been installed on the physical node and a virtual machine has been created, the virtual machine appears to be a physical node for use as application nodes.

Testimony of Dr. Shenoy (Ex. 2002) at 68:12-69:2; *see* POR at 26-30

’138 Patent



’138 Patent (Ex. 1001) at Fig. 27a-27b

’138 Patent | Three-Step Deployment Process – Figures 27A-B

’138 Patent

FIG. **27**B is a flowchart illustrating an exemplary procedure for deploying an operating system and application image instance to a node that is executing a virtual machine manager. To start the procedure, administrator **20** may instruct control node **12** to create a new tier, including defining the number of slots for the tier, or select an existing tier (**446**). Administrator **20** may then specify a captured operating system and application image for the tier (**448**). In addition, administrator **20** may set minimum, maximum, and target values for the image instances to be deployed to the slots new tier (**450**). Administrator **20** may then activate the tier (**452**). After administrator **20** activates the tier, based on the current state and goals of distributed computing system **10**, control node **12** autonomically selects a virtual machine from free pool **13**, associates the virtual machine with a slot in the tier, identifies the virtual disk image file (image instance) associated with the particular slot, copies the virtual disk image file to the local hard disk of the physical node on which the virtual node resides or the an associated SAN, and finally boots the virtual machine from the virtual disk image file.

STEP 3

’138 Patent (Ex. 1001) at 35:48-67; *see* POR at 26-30

’138 Patent

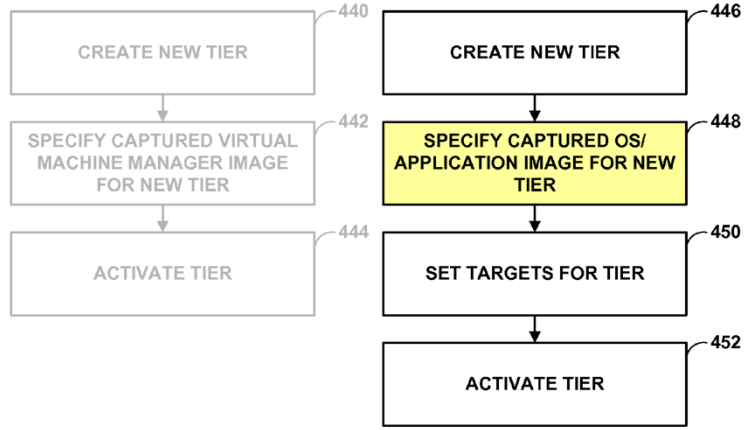


FIG. 27A

FIG. 27B

’138 Patent (Ex. 1001) at Fig. 27a-27b

’138 Patent | Three-Step Deployment Process – Figures 27A-B

Petitioner’s Expert: Dr. Shenoy

Q. Okay. And that first sentence talks about the flow chart that illustrates an exemplary procedure for deploying an operating system and application image instance to a node that is executing a virtual machine manager; correct?

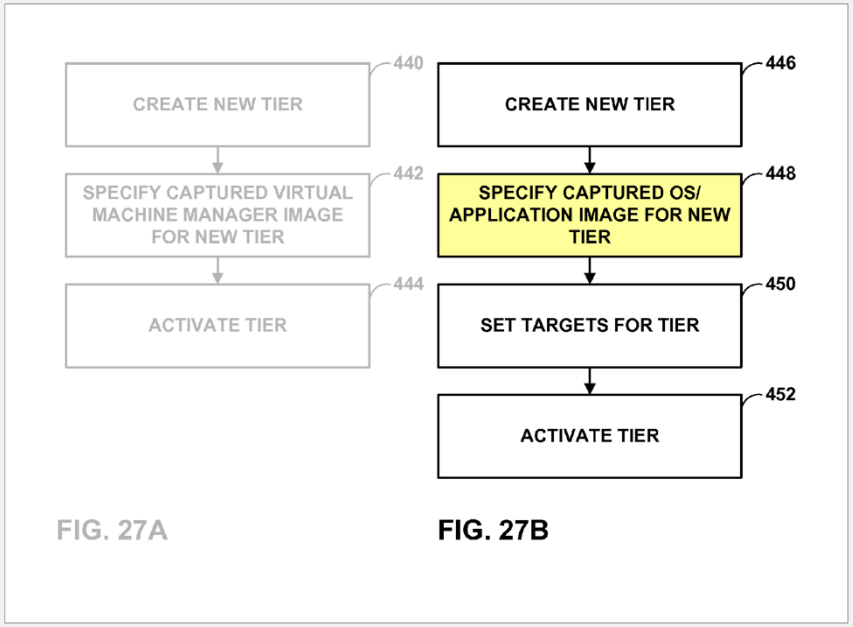
A. So this is an example procedure for deploying an OS and an application image to a node that's executing a virtual machine manager.

Q. So in this particular example, the node is running a virtual machine manager, and this flow chart is talking about a procedure for deploying the operating system and the application image instance on top of that node that's running the virtual machine manager; correct?

A. So in this example, you are deploying an OS and an application image to a node that is running a virtual machine manager.

Testimony of Dr. Shenoy (Ex. 2002) at 71:12-72:6; *see* POR at 26-30

’138 Patent



’138 Patent (Ex. 1001) at Fig. 27a-27b

’138 Patent | Three-Step Deployment Process – Figures 27A-B

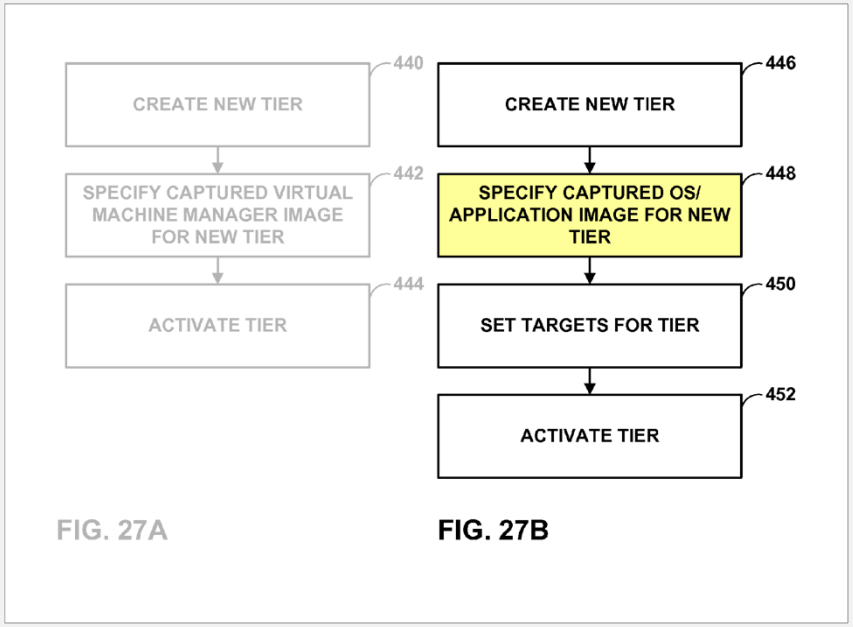
Petitioner’s Expert: Dr. Shenoy

Q. So in these examples -- or this example that we've shown in Sundar, it appears that Sundar teaches that you store the image instances of the virtual machine manager and the image instance of the applications separately so they can be deployed in two different steps; correct?

A. So certainly the virtual machine manager image and the image of the software applications are stored separately in this example on the image repository, and then they are deployed. In this example, the virtual machine manager image is deployed, and then the virtual disk image is deployed.

Testimony of Dr. Shenoy (Ex. 2002) at 73:2-74:3; *see* POR at 26-30

’138 Patent



’138 Patent (Ex. 1001) at Fig. 27a-27b

Dr. Shenoy Agrees | Three-Step Deployment Process

STEPS	'138 Patent Specification
1	The image instance of the virtual machine manager deploys before the image instance of the software applications. Ex. 2002 at 63:10-64:6; 68:12-69:2; 73:2-74:3.
2	Once the image instance of the virtual machine manager is deployed on the node, it creates the virtual machine. *Id.* at 68:12-69:2; *see* also id. at 73:2-74:3
3	The image instance of the software application (e.g., virtual disk) deploys after the deployment of the virtual machine manager and after the creation of the virtual machine. The image instance of the software application deploys onto the virtual machine (i.e., the unallocated node that is executing the virtual machine manager). *Id.* at 63:10-64:6; 73:2-74:3.

POR at 28-31

’138 Patent | Claims Require an Ordered Sequence of Steps

Claim Language requires an ordered of steps:

1. deployment of the image instance of the virtual machine manager;

2. creation of virtual machine; and

3. deployment of the image instance of the software application on the virtual machine

POR at 31-42

Claim 1 | Three-Step Deployment Process

’138 Patent

The invention claimed is:

1. A distributed computing system comprising:

a software image repository comprising non-transitory, computer-readable media operable to store: (i) a plurality of image instances of a virtual machine manager that is executable on a plurality of application nodes, wherein when executed on the applications nodes, the image instances of the virtual machine manager provide a plurality of virtual machines, each of the plurality of virtual machine operable to provide an environment that emulates a computer platform, and (ii) a plurality of image instances of a plurality of software applications that are executable on the plurality of virtual machines; and

a control node that comprises an automation infrastructure to provide autonomic deployment of the plurality of image instances of the virtual machine manager on the application nodes by causing the plurality of image instances of the virtual machine manager to be copied from the software image repository to the application nodes and to provide autonomic deployment of the plurality of image instances of the software applications on the virtual machines by causing the plurality of image instances of the software applications to be copied from the software image repository to the application nodes.

’138 Patent (Ex. 1001) at 36:25-49; *see* POR at 32-40

’138 Patent

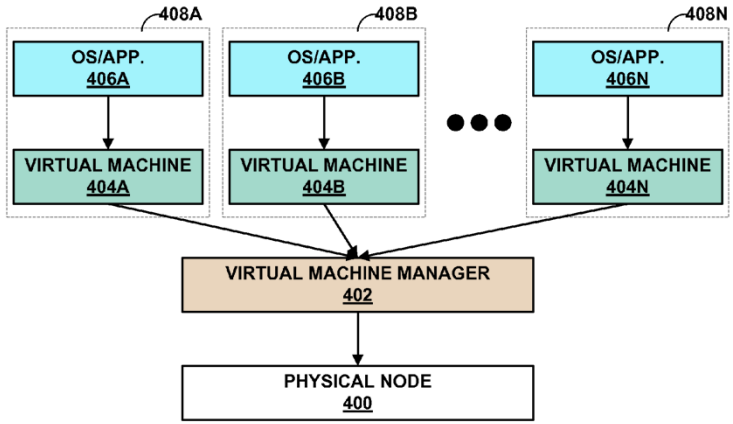


FIG. 24

’138 Patent (Ex. 1001) at Fig. 24

Claim 1 | Three-Step Deployment Process

Patent Owner's Expert: Dr. Rosenblum

83. Accordingly, my opinion is that the logic of Claim 1 requires that the control node first deploy the image of the virtual machine manager, which executes on the application node to which it is deployed, in order to create the virtual machine so that the image instances of the software applications can then be deployed: If the image instances of the virtual machine manager had not been deployed and executed, the control node would be unable to deploy any image instances of the software applications **on** the virtual machines, because no virtual machine would have been

Declaration of Rosenblum (Ex. 2001) at ¶ 83; *see* POR at 38

Claim 1 | Three-Step Deployment Process – "on the virtual machines"

'138 Patent

The invention claimed is:

1. A distributed computing system comprising:

a software image repository comprising non-transitory, computer-readable media operable to store: (i) a plurality of image instances of a virtual machine manager that is executable on a plurality of application nodes, wherein when executed on the applications nodes, the image instances of the virtual machine manager provide a plurality of virtual machines, each of the plurality of virtual machine operable to provide an environment that emulates a computer platform, and (ii) a plurality of image instances of a plurality of software applications that are executable on the plurality of virtual machines; and

a control node that comprises an automation infrastructure to provide autonomic deployment of the plurality of image instances of the virtual machine manager on the application nodes by causing the plurality of image instances of the virtual machine manager to be copied from the software image repository to the application nodes and to provide autonomic deployment of the plurality of image instances of the software applications on the virtual machines by causing the plurality of image instances of the software applications to be copied from the software image repository to the application nodes.

'138 Patent (Ex. 1001) at 36:25-49; *see* POR at 32-40

The only way to create virtual machines disclosed in the '138 Patent is for a virtual machine manager image instance to execute on an application node

POR at 22, 32-40

Petitioner's Expert: Dr. Shenoy

Q. Are you aware of anything in the Sundar patent that refers to a way that a virtual machine is created, other than being created by a virtual machine manager?

A. The two examples that we just discussed were ones where the virtual machine manager was creating the virtual machine. Those were figures 26 and 27A. I have not read the patent, but I cannot recall other disclosures that say otherwise from those two figures.

Testimony of Dr. Shenoy (Ex. 2002) at 33:12-21

Claim 1 | Three-Step Deployment Process – "on the virtual machines"

'138 Patent

nodes and to provide autonomic deployment of the plurality of image instances of the software applications on the virtual machines by causing the plurality of image

'138 Patent (Ex. 1001) at 36:45-47; see POR at 20-22, 35-40

'138 Patent

(collectively, virtual machines **404**). The term virtual machine is used herein to refer to software that creates an environment that emulates a computer platform. For example, virtual machines **40** provide environments on which guest operating systems (OS) execute as through the operating systems were operating directly on a physical computing platform. In this manner, multiple operating systems and

'138 Patent (Ex. 1001) at 32:67-33:6

'138 Patent

manager (e.g., virtual machine manager **402**) (**420**). Administrator **20** may then configure the virtual machine manager to create a virtual machine (e.g., virtual machine **404**) on node **400** (**422**). After creating the virtual machine, administrator **20** installs the particular operating system on virtual machine **404** (**424**). Administrator **20** may then install one or more

'138 Patent (Ex. 1001) at 35:6-11;

Claim 1 | Three-Step Deployment Process – "on the virtual machines"

Within the context of the Challenged Claims, a virtual machine is created when the image instance of the virtual machine manager executes on the application node. Image instances of the software applications are then executable on the virtual machines.

POR at 35-40

Petitioner's Expert: Dr. Shenoy

Q. So would you agree with me, Dr. Shenoy, that the image instances of the virtual machine manager must be capable of creating a virtual machine on the software -- I'm sorry -- on the application nodes when executed?

A. So when executed, the virtual machine manager will provide one or more virtual machines on that application node.

. . .

Q. And I might have asked a bad question, so I apologize if I did. But those image instances of the software applications that you just mentioned must be capable of being **executed on the virtual machines**. And in the context of claim 1, those **virtual machines are created when the plurality of virtual machine managers are executed**; correct?

A. So that was a long question, and I will repeat what I said earlier. That when executed on the application nodes, the virtual machine manager provides a plurality of virtual machines. And also, that the image instances of software applications are executable on the virtual machines.

Testimony of Dr. Shenoy (Ex. 2002) at 84:22-85:7; 85:18-86:11

Claim 1 | Three-Step Deployment Process – "on the virtual machines"

'138 Patent

The invention claimed is:

1. A distributed computing system comprising:

a software image repository comprising non-transitory, computer-readable media operable to store: (i) a plurality of image instances of a virtual machine manager that is executable on a plurality of application nodes, wherein when executed on the applications nodes, the image instances of the virtual machine manager provide a plurality of virtual machines, each of the plurality of virtual machine operable to provide an environment that emulates a computer platform, and (ii) a plurality of image instances of a plurality of software applications that are executable on the plurality of virtual machines; and

a control node that comprises an automation infrastructure to provide autonomic deployment of the plurality of image instances of the virtual machine manager on the application nodes by causing the plurality of image instances of the virtual machine manager to be copied from the software image repository to the application nodes and to provide autonomic deployment of the plurality of image instances of the software applications on the virtual machines by causing the plurality of image instances of the software applications to be copied from the software image repository to the application nodes.

'138 Patent (Ex. 1001) at 36:25-49; *see* POR at 32-40

Process Control Corp. v. HydReclaim Corp.

"[t]he presence of that identical language clearly indicates that **'a discharge rate'** in clause [b] is the same as **'the discharge rate'** in clause [d], both referring to the rate (in units of weight per unit of time) that material is discharged from the common hopper to the material processing machine. In addition, that conclusion avoids any **lack of antecedent basis** problem for the occurrence of 'the discharge rate' in clause [d]")

See Process Control Corp. v. HydReclaim Corp., 190 F.3d 1350, 1356-57 (Fed. Cir. 1999) (emphasis added)

Claim 19 | Requires the Same Three-Step Deployment Process

'138 Patent

19. A method comprising:

storing a plurality of image instances of a virtual machine manager that is executable on a plurality of application nodes in a distributed computing system, wherein the image instances of the virtual machine manager provide a plurality of virtual machines, each virtual machine operable to provide an environment that emulates a computer platform;

storing a plurality of image instances of a plurality of software applications that are executable on the plurality of virtual machines;

executing a rules engine to perform operations to autonomically deploy the image instances of the virtual machine manager on the application nodes by causing the image instances of the virtual machine manager to be copied to the application nodes; and

executing the rules engine to perform operations to autonomically deploy the image instances of the software applications on the virtual machines by causing the image instances of the software applications to be copied to the application nodes.

'138 Patent (Ex. 1001) at 38:1-21; *see* POR at 41-42

'138 Patent

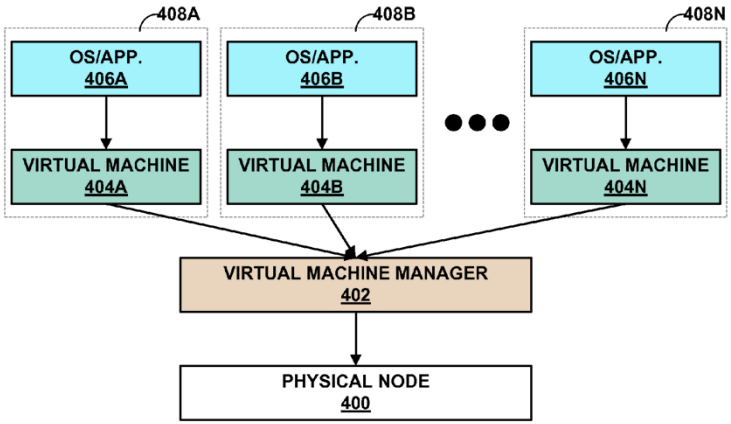


FIG. 24

'138 Patent (Ex. 1001) at Fig. 24

Claim 19 | Requires the Same Three-Step Deployment Process

Petitioner's Expert: Dr. Shenoy

Q. Okay. And the -- with respect to the storing elements of **claim 19**, the plurality of image instances of the virtual machine manager are separate images from the plurality of image instances of the plurality of software applications; correct?

A. So, again, here the plurality of image instances of the virtual machine manager are separate from the plurality of image instances in the software applications.

Q. Okay. And the plurality of image instances of the plurality of software applications must be able to be executed on a plurality of virtual machines; correct?

A. So it does say in the second storing step that the plurality of image instances of a plurality of software applications that are executable on the plurality of virtual machines. So they are executable or can be executed on the plurality of virtual machines.

Testimony of Dr. Shenoy (Ex. 2002) at 98:16-99:13; *see* POR at 41-42

Claim 26 | Requires the Same Three-Step Deployment Process

'138 Patent

26. A non-transitory, computer-readable medium comprising instructions stored thereon, that when executed by a programmable processor:

store a plurality of image instances of a virtual machine manager that is executable on a plurality of application nodes in a distributed computing system, wherein the image instances of the virtual machine manager provide a plurality of virtual machines, each virtual machine operable to provide an environment that emulates a computer platform;

store a plurality of image instances of a plurality of software applications that are executable on the plurality of virtual machines;

perform operations to autonomically deploy the image instances of the virtual machine manager on the application nodes by causing the image instances of the virtual machine manager to be copied to the application nodes; and

perform operations to autonomically deploy the image instances of the software applications on the virtual machines by causing the image instances of the software applications to be copied to the application nodes.

'138 Patent (Ex. 1001) at 38:61-67-39:1-15; *see* POR at 42

'138 Patent

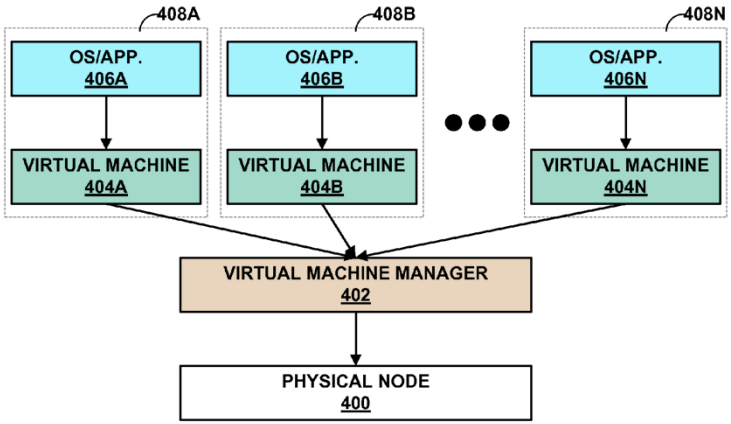


FIG. 24

'138 Patent (Ex. 1001) at Fig. 24

Petitioner's Claim Construction

Improperly Seeks to Avoid Three-Step Deployment Process

Petitioner's Construction

A. "a plurality of image instances of a plurality of software applications that are executable on the plurality of virtual machines" (Claim 1)

A POSITA would have understood this term is satisfied by, but not limited to, a plurality of virtual disk image files. Dependent claim 2 specifies that "the image instances of the plurality of software applications comprise virtual disk image files." A teaching sufficient to satisfy claim 2 must also be sufficient to satisfy claim 1. *Id.* ¶38.

Petition at 13; *see* POR at 17-22; 42-45

'138 Patent

images for storage as virtual disk image files **408**. Specifically, control node **12** may utilize the captured images as a golden images for the particular operating systems, applications and specific configuration of the virtual machine. The golden image may then be used to produce image instances, i.e., virtual disk images in this case, that are bootable on the virtual machines. The image instances may likewise be stored in application matrix **350** for autonomic deployment.

'138 Patent (Ex. 1001) at 33:60-67

Petitioner attempts to read "that are executable on the plurality of virtual machines" out of the Challenged Claims

All Challenged Claims Are Valid Over Petitioner's Asserted Prior Art

Ground 1: Khandekar

Khandekar

> The main feature of the invention is a VM provisioning server **800**, which, like any conventional computer system, will include system hardware, system software, devices, etc.

Khandekar Patent (Ex. 1004) at 9:26-27

> A provisioning server automatically configures a virtual machine (VM) according to user specifications and then deploys the VM on a physical host. The user may either choose from a list of pre-configured, ready-to-deploy VMs, or he may select which hardware, operating system and application(s) he would like the VM to have. The provisioning server then configures the VM accordingly, if the desired configuration is available, or it applies heuristics to configure a VM that best matches the user's request if it isn't. The invention also includes mechanisms for monitoring the status of VMs and hosts, for migrating VMs between hosts, and for creating a network of VMs.

Khandekar Patent (Ex. 1004) at Abstract *see* POR at 46-49

All Challenged Claims Are Valid Over Petitioner's Asserted Prior Art

Khandekar – Using VM in a different context than "virtual machine" in the '138 Patent

'138 Patent

manager (e.g., virtual machine manager **402**) (**420**). Administrator **20** may then configure the virtual machine manager to create a virtual machine (e.g., virtual machine **404**) on node **400** (**422**). After creating the virtual machine, administrator **20** installs the particular operating system on virtual machine **404** (**424**). Administrator **20** may then install one or more

'138 Patent (Ex. 1001) at 35:6-11; *see* POR at 46-49

Khandekar

A "model virtual machine" is a virtual machine (VM) that contains a certain configuration of guest operating system **320**, service pack, and applications **360**, where the applications have been tested and are known to work correctly. A model VM may be used to provision new VMs with the same configuration.

Khandekar (Ex. 1004) at 8:51-56

All Challenged Claims Are Valid Over Petitioner's Asserted Prior Art

Khandekar – Use of "VM" is different than "virtual machine" in the '138 Patent

Dr. Shenoy – describing VMs in Khandekar

Q. Okay. And do those three choices represent the three types of VMs that we just discussed?

A. The three choices here talk about the prebuilt model VM, the custom-built, and instantly provisioned, also referred to as an instant deploy VM. So those are the three examples of VMs that we were just talking about.

Q. Okay. And each of those three examples, those VMs would include at least an operating system and some applications; is that correct?

A. So all of those three would include an operating system. They would also include applications. And there are em- -- the embodiments described in the reference where it talks about custom-built VMs that start with an application only. And then that -- sorry – with an operating system only, and then the applications are installed later on.

Testimony of Dr. Shenoy (Ex. 2002) at 108:17-109:13; *see* POR at 46-49

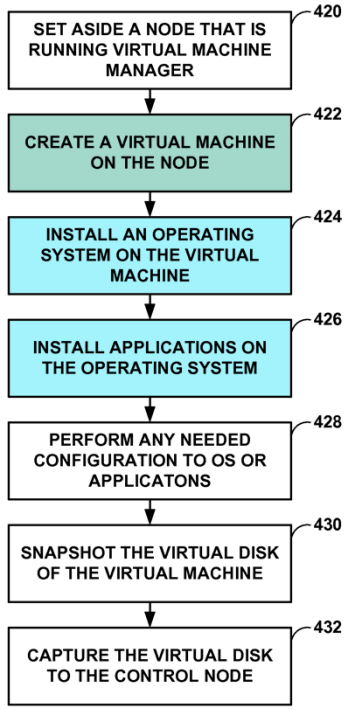


FIG. 26

'138 Patent (Ex. 1001) at Fig. 26

All Challenged Claims Are Valid Over Petitioner's Asserted Prior Art

Khandekar – Use of "VM" is different than "virtual machine" in the '138 Patent

Petitioner's Expert: Dr. Shenoy

Q. Is a virtual machine and a virtual machine image two different things?

A. So I've explained that is right in my declaration. A virtual machine image is different from a virtual machine. I've also explained that Khandekar sometimes uses the shorthand when he says VM, it's referring to the VM images, and depending on the context, sometimes the VM means the virtual machine itself.

Testimony of Dr. Shenoy (Ex. 2007) at 95:4-12; *see* Patent Owner Sur-Reply at 1-2

Dr. Shenoy acknowledged this distinction of the use of the term VM in Khandekar

All Challenged Claims Are Valid Over Petitioner's Asserted Prior Art

Khandekar – Provides little disclosure on how virtual machine monitor is deployed

Khandekar

actual, physical memory **112** and storage devices **114**. This interface is often referred to as a "virtual machine monitor" (VMM).

A VMM is usually a thin piece of software that runs directly on top of the host and virtualizes all, or at least some of, the resources of the machine, or at least of some machine.

Khandekar Patent (Ex. 1004) at 7:4-9

Khandekar

In the following description of the invention, merely for the sake of simplicity, only one VM/VMM pair is discussed. The discussion applies equally, however, to all such VM/VMM pairs that may be included in any given implementation of the invention. Moreover, it will be assumed below that when a VM is installed on a host, then a corresponding VMM is either already installed on the host, or is included and installed together with the VM.

Khandekar Patent (Ex. 1004) at 8:36-43; *see* POR at 48-49

Khandekar

As is mentioned above, a VMM is typically included for each VM in order to act as its software interface with the underlying host system. In the preferred embodiment of the invention, a VMM is already pre-installed on each host **1000**, **1001** so as to ensure compatibility and proper configuration. Several products are available from VMware, Inc. of Palo Alto, Calif., that configure conventional computers, both stand-alone desktop computers and multi-user servers, to load and run multiple VMs. It would also be possible, however, for a corresponding VMM to be included along with each staged VM, as long as the characteristics of the host on which the VM is to be installed are known. An advantage of pre-installing a VMM, without a VM, on a plurality of hosts, such as hosts **1000**, **1001**, **1003**, is that it will then be possible to deploy a given VM on any arbitrary one (or more) of these hosts. Moreover, as long as the VMM on each host exports a

Khandekar Patent (Ex. 1004) at 13:17-35

All Challenged Claims Are Valid Over Petitioner's Asserted Prior Art

Khandekar Does Not Disclose the Three Step Deployment Process

'138 Patent

The invention claimed is:

1. A distributed computing system comprising:

a software image repository comprising non-transitory, computer-readable media operable to store: (i) a plurality of image instances of a virtual machine manager that is executable on a plurality of application nodes, wherein when executed on the applications nodes, the image instances of the virtual machine manager provide a plurality of virtual machines, each of the plurality of virtual machine operable to provide an environment that emulates a computer platform, and (ii) a plurality of image instances of a plurality of software applications that are executable on the plurality of virtual machines; and

a control node that comprises an automation infrastructure to provide autonomic deployment of the plurality of image instances of the virtual machine manager on the application nodes by causing the plurality of image instances of the virtual machine manager to be copied from the software image repository to the application nodes and to provide autonomic deployment of the plurality of image instances of the software applications on the virtual machines by causing the plurality of image instances of the software applications to be copied from the software image repository to the application nodes.

'138 Patent (Ex. 1001) at 36:25-49; *see* POR at 50-52

Khandekar

"Deployment" is the step of taking a staged VM, copying it to a host machine and letting it configure with a new identity. The process of deployment is complete when the virtual machine is ready to be accessed and used by a user.

Khandekar Patent (Ex. 1004) at 8:64-67

All Challenged Claims Are Valid Over Petitioner's Asserted Prior Art

Khandekar – Does Not Disclose the Three Step Deployment Process of the Challenged Claims

Patent Owner

then deployed onto the host is a single step. Indeed, Khandekar states that the deployment process is a single step: "'Deployment' is **the step** of taking a staged VM, copying it to a host machine and letting it configure with a new identity."

POR at 50

Petitioner's Expert: Dr. Shenoy

Q. Okay. And do those three choices represent the three types of VMs that we just discussed?

A. The three choices here talk about the prebuilt model VM, the custom-built, and instantly provisioned, also referred to as an instant deploy VM. So those are the three examples of VMs that we were just talking about.

Q. Okay. And each of those three examples, those VMs would include at least an operating system and some applications; is that correct?

A. So all of those three would include an operating system. They would also include applications. And there are em- -- the embodiments described in the reference where it talks about custom-built VMs that start with an application only. And then that -- sorry – with an operating system only, and then the applications are installed later on.

Testimony of Dr. Shenoy (Ex. 2002) at 108:17-109:13; *see* POR at 47-50

All Challenged Claims Are Valid Over Petitioner's Asserted Prior Art

Khandekar – Does Not Disclose the Three Step Deployment Process of the Challenged Claims

Petitioner relies on "Fig. 1" prior art for limitation [1d]

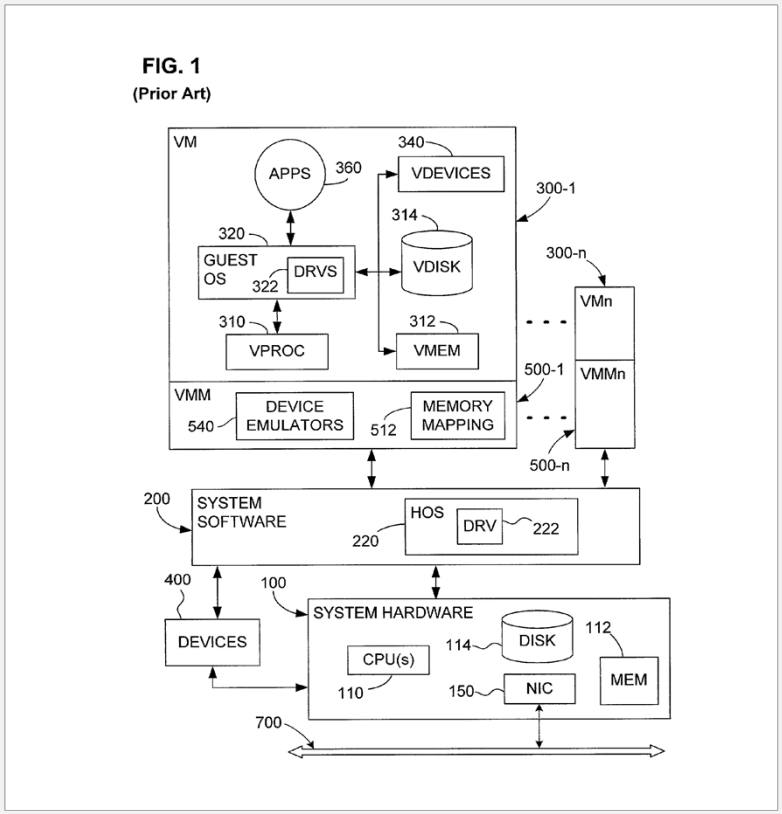
Petition

[1d] **wherein when executed on the applications nodes, the image instances of the virtual machine manager provide a plurality of virtual machines, each of the plurality of virtual machine operable to provide an environment that emulates a computer platform, and**

Khandekar renders this limitation obvious. Ex. 1003 ¶¶112-116. First, Khandekar teaches or suggests that, when executed on the applications nodes (e.g., hosts), the image instances of the virtual machine manager (e.g., VMM, for example labelled 500-1 "VMM" through 500-n "VMMn" in Fig. 1) are executed on the host computer and provide a plurality of virtual machines (e.g., labelled 300-1 "VM" through 300-n "VMn"). *See* Ex. 1004, Fig. 1:

Petition at 37; *see* POR at 49-52

Khandekar



Khandekar Patent (Ex. 1004) at Fig. 1

All Challenged Claims Are Valid Over Petitioner's Asserted Prior Art

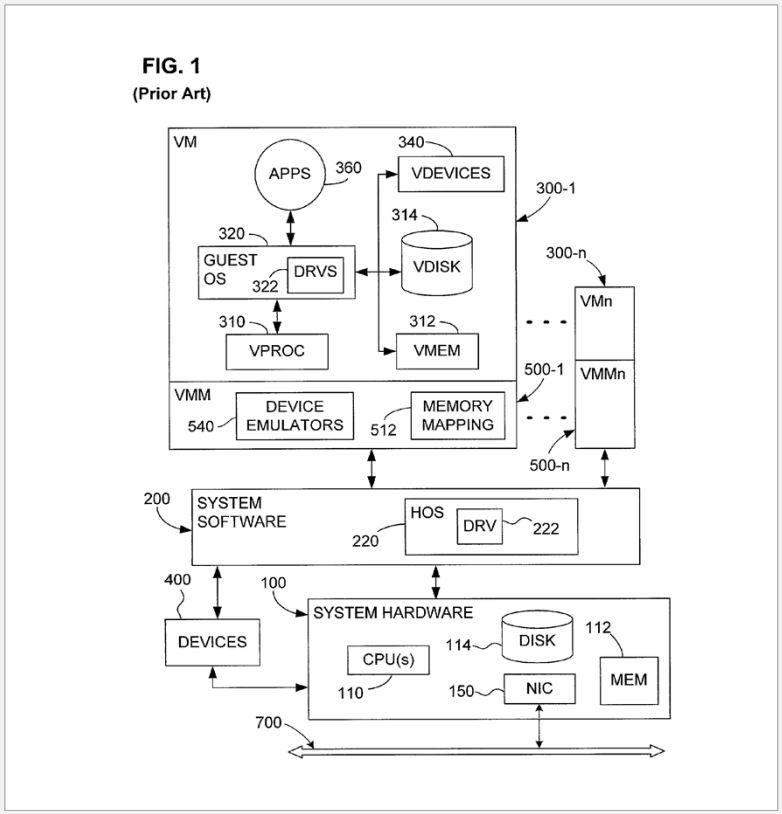
Khandekar – Does Not Disclose the Three Step Deployment Process of the Challenged Claims

Patent Owner's Expert: Dr. Rosenblum

120. I note that FIG. 1 of Khandekar is labeled "Prior Art" and thus is not Khandekar's invention. It certainly is not part of the invention disclosed in FIG. 2. In describing FIG. 1, Khandekar offers no explanation (and indeed no disclosure at all) of how the VMMs (500-1 and 500-n) might have arrived on the hosts (whether through deployment or any other method). In my opinion, therefore, Petitioner does not explain how FIG. 1 teaches that VMMs (500-1 and 500-n) can provide "a plurality of virtual machines" on the application nodes. In my opinion, element [1d] of Claim 1 is significant because it recites how the virtual machine is created. Neither FIG. 1 nor any of the accompanying description (all of which are "Prior Art" within the disclosure of Khandekar) provide any disclosure of how the disclosed VMM creates or provides the VM. It is my opinion, therefore, that Petitioner has not demonstrated that Khandekar satisfies limitation [1d].

Declaration of Rosenblum (Ex. 2001) at ¶ 120; *see* POR at 50-52

Khandekar



Khandekar Patent (Ex. 1004) at Fig. 1

All Challenged Claims Are Valid Over Petitioner's Asserted Prior Art

Khandekar – Does Not Disclose the Three Step Deployment Process of the Challenged Claims

Petitioner relies on Khandekar's disclosure related to prebuilt, custom-built, and instantly deployable VMs for limitation [1e] – Examples of which are shown in Fig. 2

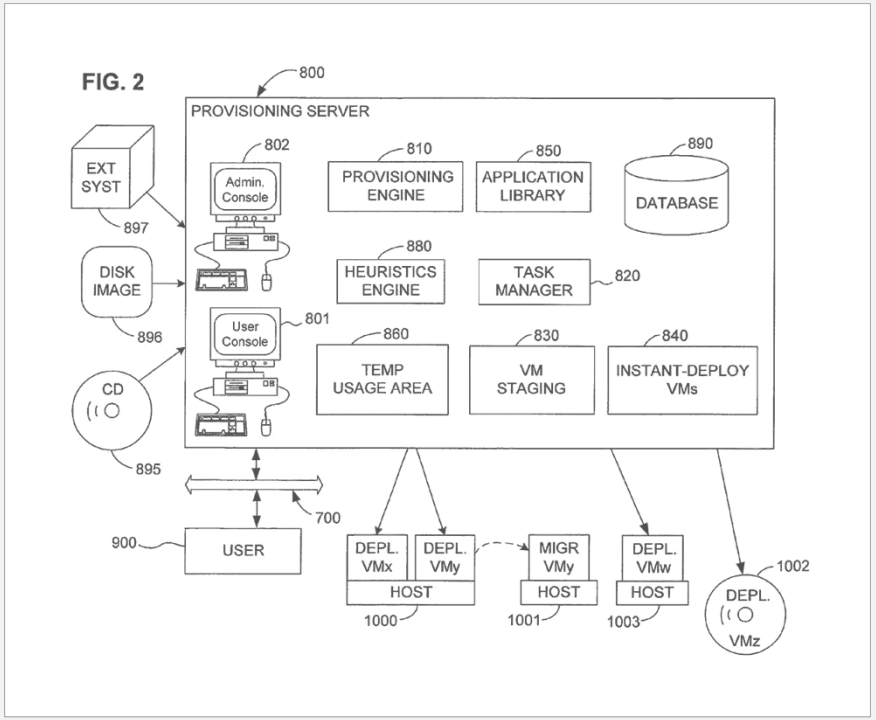
Petition

[1e]	**(ii) a plurality of image instances of a plurality of software applications that are executable on the plurality of virtual machines; and**

Khandekar teaches or suggests this limitation. *See e.g.*, Ex. 1004, 18:56-19:2, 10:15-26, 14:4-18; Ex. 1003 ¶¶122-139. Limitation [1e] may be satisfied by a plurality of virtual disk image files (*see* §V.A), which Khandekar teaches three ways: with pre-built, custom-built, and instantly-deployable VMs. Khandekar presents the user with three choices for storing and deploying image instances. *See* Ex. 1004, 9:10-25:

Petition at 41; *see* POR at 50-52

Khandekar



Khandekar Patent (Ex. 1004) at Fig. 2

All Challenged Claims Are Valid Over Petitioner's Asserted Prior Art

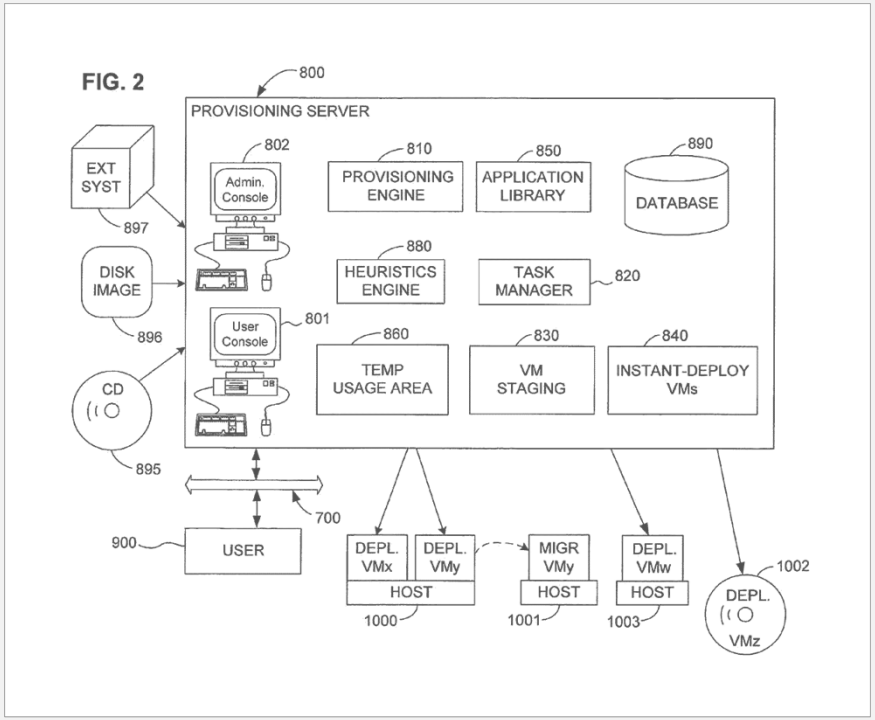
Khandekar Does Not Disclose the Three Step Deployment Process

Patent Owner's Expert: Dr. Rosenblum

135. First, with regard to the "instant deploy" VMs, the Khandekar specification explains that this disclosure does not describe deployment of an image instance at all, but rather the reallocation of already-running VMs. *See* Ex. 1004 at 9:23-25 ("Instantly provision a VM by resuming it from a suspended state. Such a VM is referred to below as a [sic] an 'instant deploy' VM."). As described by Khandekar, instant-deploy VMs are fully configured and thus do not include the step of "provid[ing] the autonomic deployment of the plurality of image instances of the software applications on the virtual machines . . . ," as required by the '138 Patent. *See id.* at 17:31-33 ("This component of the provisioning server 800 contains fully configured VMs that are suspended and are ready to be instantly deployed.").

Declaration of Rosenblum (Ex. 2001) at ¶ 135; *see* POR at 52-53

Khandekar



Khandekar Patent (Ex. 1004) at Fig. 2

All Challenged Claims Are Valid Over Petitioner's Asserted Prior Art

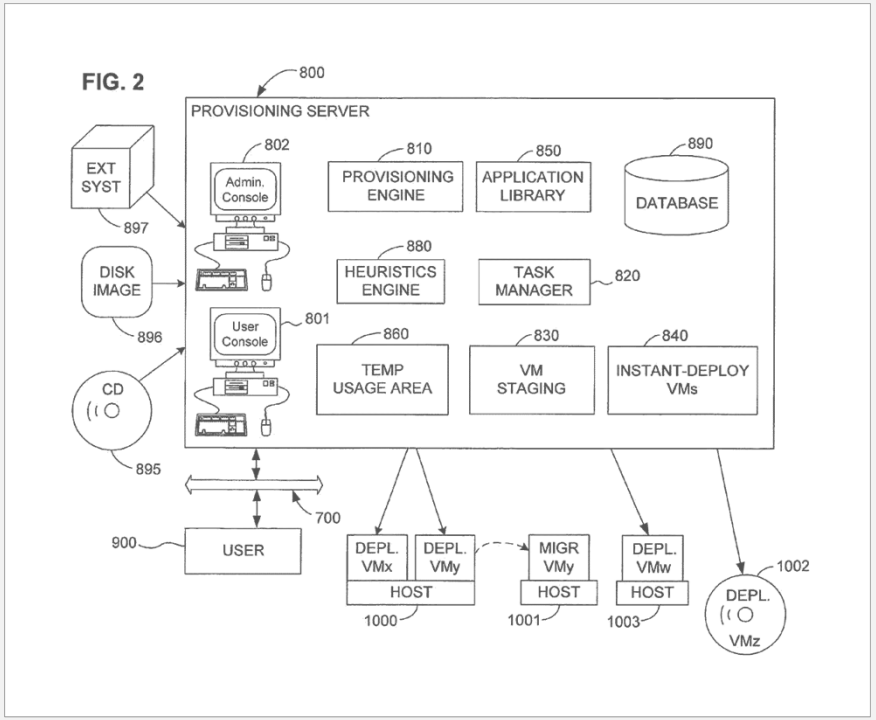
Khandekar Does Not Disclose the Three Step Deployment Process

Patent Owner's Expert: Dr. Rosenblum

136. Second, with regard to the "pre-built" VMs, Khandekar discloses that, "[i]n the preferred embodiment of the invention, pre-built VMs will be stored with their operating systems and applications installed . . . ." *Id.* at 15:43-45. This directly contradicts the Challenged Claims, which require that the software application image instances be deployed on the virtual machines as the final step of the three-stage deployment process.

Declaration of Rosenblum (Ex. 2001) at ¶ 136; *see* POR at 52-53

Khandekar



Khandekar Patent (Ex. 1004) at Fig. 2

All Challenged Claims Are Valid Over Petitioner's Asserted Prior Art

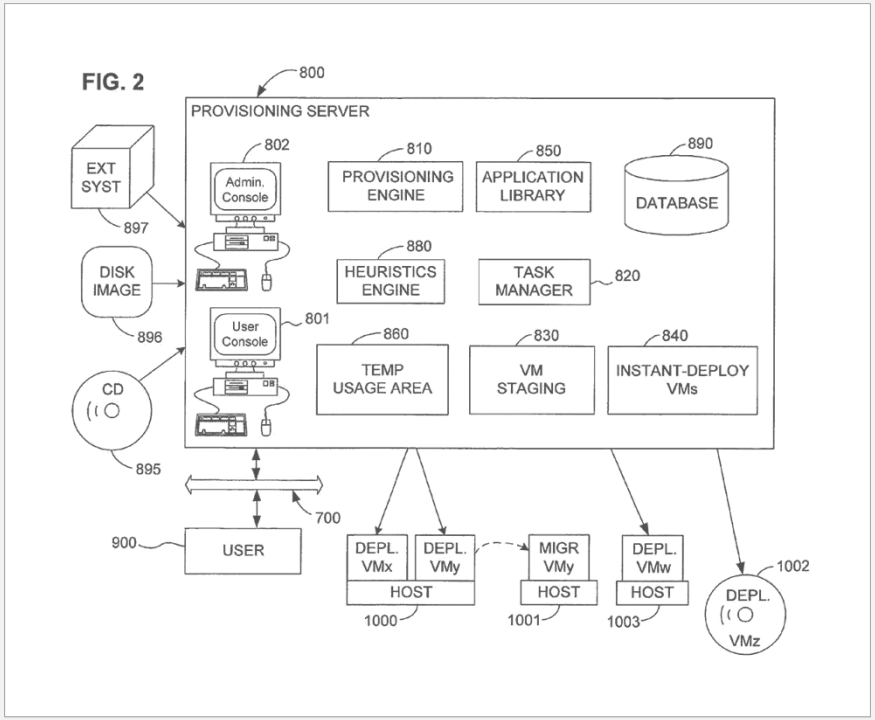
Khandekar Does Not Disclose the Three Step Deployment Process

Patent Owner's Expert: Dr. Rosenblum

137. Third, with regard to the "custom built" VMs, Khandekar discloses that "custom built VMs will be stored with just their operating systems installed." *Id.* at 15:45-46. To provision a custom built VM, Khandekar discloses that a provisioning engine "creates the custom answer file 864 and the application installation software 862." *Id.* at 21:46-50. As Khandekar explains, this "involves running an executable installation file 851 [provided by the application vendor] that takes input from a text file, called an 'answer file,' that specifies the parameters for installation for that application (for example, the folder to install it in) and the installation files/data 853 that need to be copied to the machine (here, virtual machine) that the application is to be installed on." *Id.* at 18:61-19:2. This "executable installation file," *i.e.*, the file used to install an application for the first time, is not equivalent to the "image instances of software applications" required by the '138 Patent. As Netflix admits,

Declaration of Rosenblum (Ex. 2001) at ¶ 137; *see* POR at 50-53

Khandekar



Khandekar Patent (Ex. 1004) at Fig. 2

All Challenged Claims Are Valid Over Petitioner's Asserted Prior Art

Petitioner attempts to avoid three-step deployment process by ignoring antecedent basis

Petitioner's Expert: Dr. Shenoy

Q. Are you familiar with the legal term "antecedent basis"?

A. No, I am not.

Q. Okay. But you would agree with me that the first time that the term "virtual machines" appears in claim 1 of the Sundar patent is 33 where it says "a plurality of virtual machines"; correct?

A. Yeah. If you ignore the words "virtual machine" in the context of the virtual machine manager that's also in the claim, the first time the words appear is in line 33 where it talks about a plurality of virtual machines.

Q. Okay. So when it says in line 40 – the very end of line 46 and the beginning of line 47, where it says "on the virtual machines," doesn't "the virtual machines" there refer to the virtual machines that were first described in line 33 of claim 1 of the Sundar patent?

A. So I haven't given an opinion on that issue in my declaration. You know, I would have to think about whether that term "virtual machine" in that sentence is referring to the plurality of virtual machines, because the previous term talks about plurality of virtual machines on many different nodes.

Q. As you sit here today, are you aware of any reason why a person of ordinary skill reading claim 1 of the Sundar patent wouldn't consider the phrase "the virtual machines" in line 47 of column 36 to mean the same thing as "a plurality of virtual machines" in line 33 of column 36 of the Sundar patent?

A. So, again, I did not opine on that particular issue. The first part of the claim talks about a software image repository. It talks about images in that repository that, when executed, are executable on a plurality of virtual machines. The second part of the claim talks about a control node that is performing deployment. So I do not have an opinion on that issue at this point.

* * *

Q. Okay. And so you -- and just to be clear, you did not offer an opinion on that issue in your declaration, which has been marked as Exhibit 1 in this deposition; correct?

A. So as far as the issue of the words "virtual machines" that appear on line 47, whether they refer to the same virtual machines that are mentioned in line 33, I have not offered an opinion in my declaration.

Testimony of Dr. Shenoy (Ex. 2002) at 90:17-92:15; 93:3-12; *see* POR at 53-56

All Challenged Claims Are Valid Over Petitioner's Asserted Prior Art

Petitioner attempts to mix and match different prior art disclosures

Patent Owner's Expert: Dr. Rosenblum

123. As such, there really is no dispute that Khandekar fails to disclose or suggest the three-step deployment process that, as I discuss above, is required by the Challenged Claims. *See supra* Section VIII. Petitioner does not argue otherwise. Instead, Petitioner selects different portions of the Khandekar reference to apply against each limitation (or each part of a limitation) of the Challenged Claims in isolation, without regard to the overall claim structure. As I noted, I do not believe that the logic or grammar of the Challenged Claims (including the principle of antecedent basis) permits this piecewise analysis. Nor do I believe that the ordering required by the specification permits the three steps of the three-step deployment process to be analyzed outside of their context together in the Challenged Claims.

Declaration of Rosenblum (Ex. 2001) at ¶ 123; *see* POR at 56

All Challenged Claims Are Valid Over Petitioner's Asserted Prior Art

Khandekar Fails to Disclose Elements [1c] and [1d] of the Challenged Claims

'138 Patent

The invention claimed is:

1. A distributed computing system comprising:

a software image repository comprising non-transitory, computer-readable media operable to store: (i) a plurality of image instances of a virtual machine manager that is executable on a plurality of application nodes, wherein when executed on the applications nodes, the image instances of the virtual machine manager provide a plurality of virtual machines, each of the plurality of virtual machine operable to provide an environment that emulates a computer platform, and (ii) a plurality of image instances of a plurality of software applications that are executable on the plurality of virtual machines; and

a control node that comprises an automation infrastructure to provide autonomic deployment of the plurality of image instances of the virtual machine manager on the application nodes by causing the plurality of image instances of the virtual machine manager to be copied from the software image repository to the application nodes and to provide autonomic deployment of the plurality of image instances of the software applications on the virtual machines by causing the plurality of image instances of the software applications to be copied from the software image repository to the application nodes.

1c

1d

'138 Patent (Ex. 1001) at 36:25-49; *see* POR at 57-60

Khandekar

As is mentioned above, a VMM is typically included for each VM in order to act as its software interface with the underlying host system. In the preferred embodiment of the invention, a VMM is already pre-installed on each host **1000**, **1001** so as to ensure compatibility and proper configuration. Several products are available from VMware, Inc. of Palo Alto, Calif., that configure conventional computers, both stand-alone desktop computers and multi-user servers, to load and run multiple VMs. It would also be possible, however, for a corresponding VMM to be included along with each staged VM, as long as the characteristics of the host on which the VM is to be installed are known. An advantage of pre-installing a VMM, without a VM, on a plurality of hosts, such as hosts **1000**, **1001**, **1003**, is that it will then be possible to deploy a given VM on any arbitrary one (or more) of these hosts. Moreover, as long as the VMM on each host exports a

Khandekar Patent (Ex. 1004) at 13:17-35

All Challenged Claims Are Valid Over Petitioner's Asserted Prior Art

Khandekar Fails to Disclose Elements [1c] and [1d] of the Challenged Claims

Patent Owner's Expert: Dr. Rosenblum

140. I believe that this teaching of Khandekar is further reinforced by the fact that Khandekar teaches that VMMs and VMs are always to be considered in pairs:

> In the following description of the invention, merely for the sake of simplicity, **only one VM/VMM pair** is discussed. The discussion applies equally, however, to **all such VM/VMM pairs that may be included** in any given implementation of the invention. Moreover, it will be assumed below that when a VM is installed on a host, **then a corresponding VMM is either already installed on the host, or is included and installed together with the VM**.

Declaration of Rosenblum (Ex. 2001) at ¶ 140; *see* also POR at 57-60

> **such VM/VMM pairs that may be included** in any given implementation of the invention. Moreover, it will be assumed below that when a VM is installed on a host, **then a corresponding VMM is either already installed on the host, or is included and installed together with the VM**.

Id. at 8:36-43 (emphasis added). Khandekar thus teaches that any VMM included with a VM is included in such a way that the VMM and VM are bound together as a pair. Furthermore, logic and grammar dictate that the phrase "included and installed together" quoted above must be read as "*included together* and installed together", thereby teaching that whenever a VMM is included with a VM, it is *included together with the VM in the same image*, since the only VMs described by Khandekar as being "installed" on a host are those VMs that have been captured in an image. This view of a VMM and VM being bound in pairs is further reinforced by Khandekar's teaching that "from the perspective of the system designer **the VM and VMM can be considered to form a single cooperating system.**" *Id.* at 7:40-42 (emphasis added). In other words, a VM/VMM pair forms a single cooperating system.

All Challenged Claims Are Valid Over Petitioner's Asserted Prior Art

Khandekar Fails to Disclose Elements [1c] and [1d] of the Challenged Claims

Petitioner's Expert: Dr. Shenoy

taught by Khandekar. *See id.* A POSITA would have found it obvious to follow these teachings from Khandekar, with VMM images being stored in the provisioning server and provided to hosts together with VM images, because doing

Declaration of Dr. Shenoy (Ex. 1003) at 51-52

Patent Owner's Expert: Dr. Rosenblum

141. In light of these teachings of Khandekar, I believe that not only is it unfounded speculation for the Petitioner to assert that a POSITA would store images of a VMM separately from images of a VM, but in order to do so, the POSITA must perform a creative step that *contradicts* the teachings of Khandekar. This creative step – the storage of images of a virtual machine manager for subsequent autonomic deployment – is disclosed only in the '138 Patent.

Declaration of Rosenblum (Ex. 2001) at ¶ 141; *see* POR at 57-60

All Challenged Claims Are Valid Over Petitioner's Asserted Prior Art

Khandekar Fails to Disclose Elements [1c] and [1d] of the Challenged Claims

Patent Owner's Expert: Dr. Rosenblum

130. In my opinion, Petitioner identifies no disclosure in any prior art, including the references cited as part of the grounds for review (Khandekar, Le (Ex. 1005), Sidebottom (Ex. 1006), and Stone (Ex. 1007)), as well as other references mentioned in the Petition (such as Powell (Ex. 1014) and French (Ex. 1016)), that suggest storing an image of a virtual machine manager in a software repository, let alone storing a plurality of such images. Petitioner quotes the Powell reference as teaching that a hypervisor can be loaded as part of an operating system or firmware image. *See* Pet. at 20 (quoting Ex. 1014, ¶ 0062). I find nothing in this disclosure that suggests virtual machine manager images might be stored on a software repository. Similarly, I find Petitioner's reliance on French's teaching that software repository. Similarly, I find Petitioner's reliance on French's teaching that a "bootable core and the virtual machine manager image must be installed in a secure partition," to be unavailing; once again, Petitioner identifies nothing in the reference that suggests this image might be stored on a software repository. *Id.* (quoting Ex. 1016, ¶ 0049). Moreover, I note that neither of these references was listed as part of the grounds asserted against the Challenged Claims.

Declaration of Rosenblum (Ex. 2001) at ¶ 130; see POR at 57-60

Petitioner cites to no evidence to support position that it would be obvious to store separate images of a virtual machine manager

All Challenged Claims Are Valid Over Petitioner's Asserted Prior Art

Khandekar Fails to Disclose Elements [1c] and [1d] of the Challenged Claims

Patent Owner's Expert: Dr. Rosenblum

131. Further, even assuming (contrary to my opinion) that a POSITA might have found it obvious to store an image of a virtual machine manager in a software repository, I see no evidence in the record that image instances of such a virtual machine manager might be capable of being executed on an application node to create (provide) a virtual machine, in the sense Khandekar uses that term. Again, in the context of the Challenged Claims, a virtual machine is a bare metal platform onto which operating systems and applications can be deployed, but it does not include the software itself. *See supra* ¶¶ 29, 52. Petitioner identifies nothing in Khandekar that suggests that an image of a virtual machine manager stored in a software repository might have such a capability. Rather, as I note above, the VMs deployed (in a single step) according to Khandekar's teachings already include an operating system and applications. *See supra* ¶ 114; *accord infra* ¶¶ 132-37. Thus, these VMs cannot also serve as the virtual machines provided/created by execution of the virtual machine manager on an application.

Declaration of Rosenblum (Ex. 2001) at ¶ 131; *see* POR at 57-60

All Challenged Claims Are Valid Over Petitioner's Asserted Prior Art

Khandekar Fails to Disclose Element [1e] of the Challenged Claims

'138 Patent

The invention claimed is:

1. A distributed computing system comprising:

a software image repository comprising non-transitory, computer-readable media operable to store: (i) a plurality of image instances of a virtual machine manager that is executable on a plurality of application nodes, wherein when executed on the applications nodes, the image instances of the virtual machine manager provide a plurality of virtual machines, each of the plurality of virtual machine operable to provide an environment that emulates a computer platform, and (ii) a plurality of image instances of a plurality of software applications that are executable on the plurality of virtual machines; and

1c

1d

1e

a control node that comprises an automation infrastructure to provide autonomic deployment of the plurality of image instances of the virtual machine manager on the application nodes by causing the plurality of image instances of the virtual machine manager to be copied from the software image repository to the application nodes and to provide autonomic deployment of the plurality of image instances of the software applications on the virtual machines by causing the plurality of image instances of the software applications to be copied from the software image repository to the application nodes.

'138 Patent (Ex. 1001) at 36:25-49; *see* POR at 60-62

Patent Owner

Section IV.E.1.a). The VMs disclosed in Khandekar already contain an operating system and/or software application at the time they are deployed by the provisioning server. *Id.* Therefore, Petitioner relies on the VMs disclosed in Khandekar to satisfy both the "image instances of the plurality of software applications" and "virtual machine" portions of this claim element. *See* Pet. at 41-47 (citing exclusively to

POR at 60

Khandekar

A "model virtual machine" is a virtual machine (VM) that contains a certain configuration of guest operating system **320**, service pack, and applications **360**, where the applications have been tested and are known to work correctly. A model VM may be used to provision new VMs with the same configuration.

Khandekar Patent (Ex. 1004) at 8:51-56

All Challenged Claims Are Valid Over Petitioner's Asserted Prior Art

Khandekar Fails to Disclose Element [1e] of the Challenged Claims

'138 Patent

The invention claimed is:

1. A distributed computing system comprising:

a software image repository comprising non-transitory, computer-readable media operable to store: (i) a plurality of image instances of a virtual machine manager that is executable on a plurality of application nodes, wherein when executed on the applications nodes, the image instances of the virtual machine manager provide a plurality of virtual machines, each of the plurality of virtual machine operable to provide an environment that emulates a computer platform, and (ii) a plurality of image instances of a plurality of software applications that are executable on the plurality of virtual machines; and

1c

1d

1e

a control node that comprises an automation infrastructure to provide autonomic deployment of the plurality of image instances of the virtual machine manager on the application nodes by causing the plurality of image instances of the virtual machine manager to be copied from the software image repository to the application nodes and to provide autonomic deployment of the plurality of image instances of the software applications on the virtual machines by causing the plurality of image instances of the software applications to be copied from the software image repository to the application nodes.

1e

'138 Patent (Ex. 1001) at 36:25-49; *see* POR at 60-62

Patent Owner

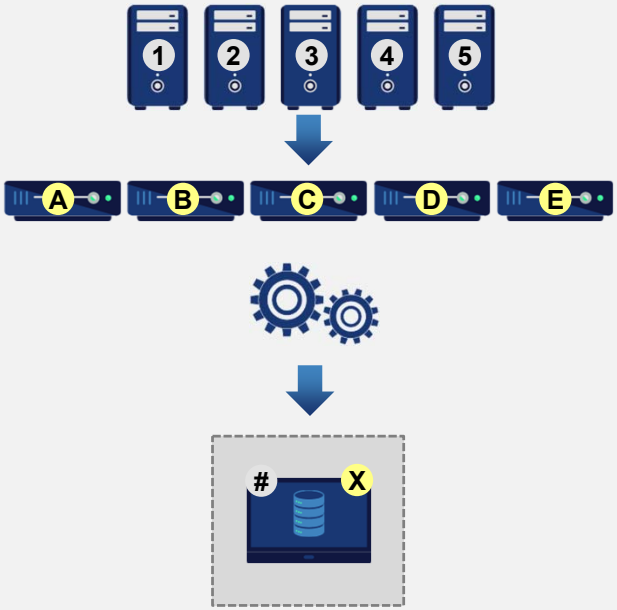
Once again, this claim element requires both "virtual machines" created by the execution of a virtual machine manager and "image instances of software applications" that deploy "on the virtual machines" – the VMs disclosed in Khandekar cannot be both. *See* Ex. 2001, ¶¶ 131-32.

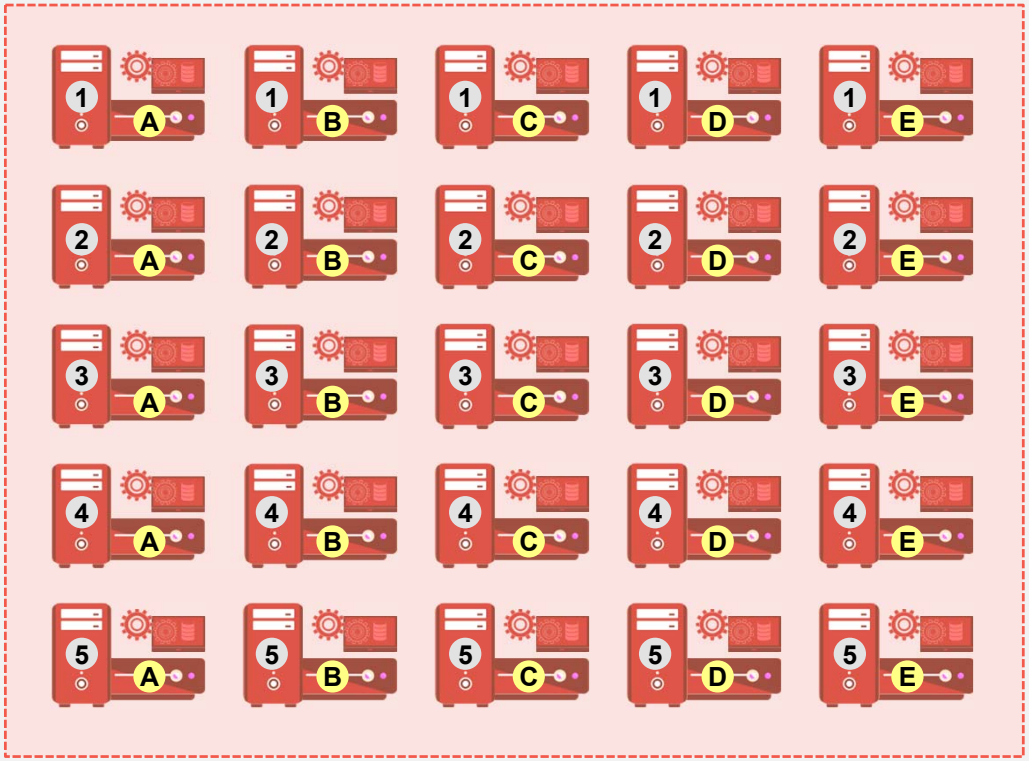
POR at 61

All Challenged Claims Are Valid Over Petitioner's Asserted Prior Art

Advantages of the Challenged Claims Over Khandekar

Example: 5 different image instances of virtual machine manager and 5 different image instances of software applications





POR at 61-64

All Challenged Claims Are Valid Over Petitioner's Asserted Prior Art

Ground 2: Khandekar in View of Le

Le adds nothing to Khandekar with respect to the Challenged Claims

Le discloses whole disk imaging with VMM software already installed

'680 Patent

Virtual disks and conventional disk images are similar in that they encapsulate the state of a computer's disk. Cloning a virtual machine, however, is generally much easier than the equivalent image deployment operation on physical computers. In order to clone a virtual machine, a user needs simply to make copies of its configuration and virtual disk files, and place them on the host computer of choice. To power on and run the cloned virtual machine, all that the host computer needs is to have the appropriate VMM software installed.

Le (Ex. 1005) at 14:22-32; POR at 64-65

Petition

Le teaches "us[ing] disk imaging to quickly provision a bare-metal computer on the network ***with a complete software stack consisting of an operating system and a set of core applications***." Ex. 1005, 54:14-21, 15:63-65, 16:29-30, 8:10-23, 22:32-44, 67:31-40. "Before the initial image is captured, the base computer is configured with an operating system and common set of software applications that are required on all clones. Any computer deployed from this image would inherit the same set of software." *Id.*, 10:55-62; Ex. 1003 ¶¶100-101.

Petition at 34; Patent Owner Sur-Reply at 21-22

All Challenged Claims Are Valid Over Petitioner's Asserted Prior Art

Ground 2: Khandekar in View of Le

Le adds nothing to Khandekar with respect to the Challenged Claims

Le discloses whole disk imaging with VMM software already installed

'680 Patent

A computer management system can use disk imaging to quickly provision a bare-metal computer on the network with a complete software stack consisting of an operating system and a set of core applications. The resulting cloned computer can then be further customized and personalized using the customization techniques described earlier. A single template image, or a small set of core template images, is used to create clones.

Le (Ex. 1005) at 54:14-21

Patent Owner's Expert: Dr. Rosenblum

150. In other words, Le, like Khandekar, presumes that a VMM has already been installed when discussing deployment of disk images, and that reference provides no disclosure of how the pre-existing installation of a VMM might have occurred. Le also appears to teach that the "cloned virtual machine," which would correspond to the software application image instance, is copied to the host before the VMM is executed. *See id.* at 54:14-21.

Declaration of Rosenblum (Ex. 2001) at ¶ 150; POR at 64-65

All Challenged Claims Are Valid Over Petitioner's Asserted Prior Art

Ground 2: Khandekar in View of Le

Implicitly acknowledging that Khandekar and/or Le do not disclose the three-step deployment process of the Challenged Claims, Petitioner asserts a new argument in its Reply

Petitioner's Expert: Dr. Shenoy

PO attacks Khandekar alone but did not address the combination of Khandekar with Le. Petitioner and Dr. Shenoy explained how the combination teaches PO's three-stage deployment process by deploying the *physical* image of a host computer (including the VMM) using Le's teachings, booting the host computer, and then deploying the *virtual* image of a VM to the host using Khandekar's teachings. This evidence stands unrebutted.

Petitioner's Reply at 21

Patent Owner

Petitioner never made this argument in the Petition. In fact, the terms "physical image" and "booting the host computer" are found nowhere in the Petition.[7] This is

Patent Owner's Sur-Reply at 21

Dr. Shenoy could only identify his "reply dec." as disclosing this position

Testimony of Dr. Shenoy (Ex. 2007) at 133:15-140:15; Patent Owner's Sur-Reply at 21

All Challenged Claims Are Valid Over Petitioner's Asserted Prior Art

Ground 2: Khandekar in View of Le

Petitioner's new argument still lacks merit

Le

A powered-off (i.e., inactive) virtual machine generally comprises a configuration file that describes the VM's set of hardware devices, such as memory size and input/output ports, and a virtual disk file. Those two files define a complete computer, and can be moved or copied from one host computer to another. Virtual machines can thus be viewed as mobile computers, totally encapsulated and represented by a set of files.

Le (Ex. 1005) at 14:14-21

Patent Owner

file." EX-1005, 14:14-17. According to this teaching, the virtual machine manager cannot create a virtual machine until the VM image has been deployed. Thus, even on this novel theory, one would have to (1) deploy the virtual machine manager image and (2) deploy the VM image before the virtual machine manager could create a virtual machine, as required by the Challenged Claims. Even this alleged combination does not include another step of deploying a software application image on to this virtual machine, as the Challenged Claims also require.

Patent Owner's Sur-Reply at 21-22

All Challenged Claims Are Valid Over Petitioner's Asserted Prior Art

Ground 3: Khandekar in View of Le and Sidebottom

Patent Owner

3. Ground 3: Khandekar, Le, and Sidebottom

Petitioner further allege that Claims 9 and 10 are obvious under 35 U.S.C. § 103, over the combination of Khandekar, Le, and Sidebottom. *See* Pet. at 9. Petitioner does not allege that Sidebottom provides any material disclosure with regard to Claims 1, 19, and 26. *See* Ex. 2001, ¶ 153-54. Those claims, therefore, are valid over the combination of Khandekar, Le, and Sidebottom for the reasons discussed *supra* Sections IV.E.1 and IV.E.2 with regard to Grounds 1 and 2. *See id.* Claims 9 and 10 are valid over the same combination at least by virtue of their dependence from Claim 1.

POR at 65

All Challenged Claims Are Valid Over Petitioner's Asserted Prior Art

Ground 4: Khandekar in View of Le and Stone

Patent Owner

4. Ground 4: Khandekar, Le, and Stone

Petitioner further allege that Claim 17 is obvious under 35 U.S.C. § 103, over the combination of Khandekar, Le, and Stone. *See* Pet. at 9. Petitioner does not allege that Stone provides any material disclosure with regard to Claims 1, 19, and 26. *See* Ex. 2001, ¶ 153-54. Those claims, therefore, are valid over the combination of Khandekar, Le, and Stone for the reasons discussed *supra* Sections IV.E.1 and IV.E.2. with regard to Grounds 1 and 2. Claim 17 is valid over the same combination at least by virtue of its dependence from Claim 1.

POR at 65-66

United States Patent and Trademark Office

Patent Trial and Appeal Board

NETFLIX, INC.,

(Petitioner)

v.

AVAGO TECHNOLOGIES INTERNATIONAL SALES

(Patent Owner)

IPR 2020-01582 | U.S. Patent No. 8,572,138

Patent Owner's Demonstrative Exhibits

ADSERO IP