Trials@uspto.gov
571-272-7822

Paper 31
Entered: January 26, 2022

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.

IPR2020-01582
Patent 8,572,138 B2

Record of Oral Hearing
Held Virtually: Wednesday, January 5, 2022

Before KRISTEN L. DROESCH, KERRY BEGLEY, and
NATHAN A. ENGELS, *Administrative Patent Judges*.

A P P E A R A N C E S

ON BEHALF OF THE PETITIONER:

JEFFREY LIANG, ESQUIRE
SHEPPARD, MULLIN, RICHTER & HAMPTON LLP
Leesburg, Virginia 20176
(703) 777-5700

ON BEHALF OF THE PATENT OWNER:

DANIEL S. YOUNG, ESQUIRE
ADSERO IP LLC d/b/a
SWANSON & BRATSCHUN LLC
8210 Southpark Terrace
Littleton, Colorado 80120
(303) 268-0066

The above-entitled matter came on for hearing on Wednesday, January 5, 2022, commencing at 1:00 p.m. EST, by video.

P R O C E E D I N G S

JUDGE DROESCH: Good afternoon. My name is Judge Droesch. Joining me for this hearing are Judge Begley and Judge Engels. We're here for the oral hearing for IPR2020-01582 where Netflix challenges claims of Avago's Patent 8,572,138.

Per the order regarding the oral hearing, petitioner is allotted one hour for its presentation. Petitioner will go first, petitioner may reserve some of its time for rebuttal, but that time cannot exceed 30 minutes, and following petitioner, patent owner will make its presentation and will also have one hour. Patent owner may also reserve some of its time for rebuttal that may not exceed 30 minutes.

Before we begin, I wanted, you know, make a couple of reminders. To aid in the preparation of the transcript, please identify yourself before speaking and identify your demonstrative slides by the slide number.

Counsel for petitioner, would you like to reserve some time for rebuttal?

MR. LIANG: Yes, Your Honor. This is Jeffrey Liang, I would like to reserve 20 minutes for rebuttal, so I'll go 40 minutes on opening and reserve 20 for rebuttal.

JUDGE DROESCH: Okay. Thank you.

All right. Counsel for petitioner, you may begin your presentation whenever you're ready.

MR. LIANG: Thank you, Your Honor. Petitioner Slide 6, and this is Jeff Liang. Petitioner Slide 6. There are three remaining disputes, so today, first, I'm going to address claim construction; second, I'll talk about virtual machines; and then third, I'll talk about the images of a virtual machine manager.

This is a straightforward case, because for all three of these remaining issues, the evidence in the record including the 138 patent, the prior art testimony from both experts, and even patent owner's own prior briefing, it all supports the petitioner.

Let's go to Slide 7. Now, the petition is supported by the testimony of Dr. Shenoy, and I just want to point out he's not just any expert. Dr. Shenoy is the co-author of prior art that's cited by the applicant on the face of the 138 patent.

As Dr. Shenoy explained, the challenged claims are invalid over Khandekar and Le.

Slide 8; so let's start with claim construction.

Slide 9; while patent owner insists that

the claims must be construed using a three-step process, the patent owner took a different position in District Court just one month before it filed the patent owner response. The patent owner did not propose any claim terms for construction, it didn't propose any constructions, and it just said no construction is necessary, because the claim terms should be understood based on their plain and ordinary meaning.

Slide 10; but now patent owner argues that Claim 1 should not be read using plain ordinary meaning, and instead ask the Board to import a three-step process into the claims. And these three steps appeared nowhere in the claim as written. Steps one and three require actual deployment of images, which is not, for example, mentioned in Claim 1, and step two requires actual execution of the manager, which is again not recited in Claim 1. So this is not a question about the ordering of steps in a claim, because these steps do not appear in the claim. The threshold question is whether these steps should be imported into the claim to begin with.

Slide 11; patent owner did not identify any term that requires construction. It didn't point to any ambiguous claim language or any claim

term that needs a definition; instead, patent owner just asked the Board to inject its three new steps somewhere into the claims.

JUDGE BEGLEY: This is Judge Begley. On that note, the patent owner response on, I believe it's pages 32 to 33, it does bold what appears to be relevant claim language of Claim 1 within its claim construction analysis. Is your position that that's not sufficient to identify the language that patent owner is asking the Board to interpret?

MR. LIANG: Correct. Your honor, it's not sufficient, because it is simply, you know, that language has no dispute about what that language means. Patent owner is trying to say -- trying to use that to say that, well, maybe there should be these extra steps, as well, but there's in terms of what the claim mean, the plurality of virtual machines, I mean, it's clear on its face that doesn't require construction, and we could just jump there to look at that.

JUDGE BEGLEY: Sure.

MR. LIANG: Let's go to slide -- well, let's go to Slide 13. You know, before the patent owner argues that this three-step process is required by the claim, and as part of that process, the virtual machine manager must provide virtual

machines. It must actually manage one or more virtual machines.

But patent owner took a contrary position in District Court where it stated that the virtual machine manager is a program that is capable of managing one or more virtual machines, not a program that actually does so, and this is an excerpt from their brief. This is patent owner's emphasis, not ours.

So patent owner has emphasized that Claim 1 is about capability, and it's cited case law in support.

In slide 14, after explaining that Claim 1 is about capability, patent owner cited the Chrimar case, which held that it's hardly uncommon for an apparatus or system claim as a claim to a product rather than a process or a forbidden mix to be directed to capability instead of actual operation.

And so Claim 1 is about capability. It's about a prop -- it's about a system, not a process. It's about capabilities rather than actual operation.

In slide 15, patent owner's expert admitted this. Patent owner's expert admitted that Claim 1 doesn't require anything to be executed. It describes what happens when the virtual machine

manager is executed.

JUDGE BEGLEY: This is Judge Begley again. Just to jump in. Your briefing, but also these slides are referring to the parties' briefing before the District Court on claim construction, which we -- I know we have in the record the briefs, the claim construction briefs from the District Court case. Has the District Court actually construed any of the terms of the 138 patent or addressed the order-of-steps interpretation issue that patent owner is raising here?

MR. LIANG: Well, the District Court has not addressed the order of steps, because that was not proposed by the patent owner in the District Court's briefing. It proposed new constructions in District Court. So that definitely was not addressed. In terms of whether the order has issued, let me check on that and get back to you on that. And if it has, we can file that if the Board likes.

JUDGE BEGLEY: That would be helpful. I think having the briefing, but not knowing what happened at the District Court is not as helpful.

MR. LIANG: Yes, Your Honor. So I will look into that and address it on rebuttal.

JUDGE BEGLEY: Thank you.

MR. LIANG: Okay. So in Slide 16, if we look at Claim 1 itself, it states in the first highlight that the VMM images are executable, meaning they're capable of being executed. And then Claim 1 says that when executed on the application nodes, the image instances provided plurality of virtual machines.

So this is akin to the Intel case shown below, which also describes what happens -- in the claim in that case also describes what happens when something occurs. And the Federal Circuit held that when the claim clearly states when a mode is selected, then the accused device need only be capable of operating in that mode. So this is about capability.

Slide 17. And so for Claim 1 focuses on capability, and there's no dispute that when that this first highlighted text shown on Slide 17, that in this first highlighted texts, the claim recites a capability when the VMM images are executed, and there's no dispute that it provides the antecedent basis for the plurality of virtual machines that are mentioned elsewhere in the claim.

So there -- the VMM images have the capability when executed to provide a plurality of virtual machines. And then later in the claim,

wherever the claim recites the plurality of virtual machines, it simply refers back to those virtual machines that the VMM images are capable of providing. So the claim is clear on its face as written. There's no terms that require construction, and that's all the Board needs to reject patent owner's construction.

Slide 18; patent owner argues you need to resort to logic to figure out how the control node provides autonomic deployment, but that's not right, because the claim provides a definition for both types of images. It says to provide autonomic deployment by causing the plurality of image instances to be copied to the nodes, and patent owner has no answer for these express definitions in the claim.

Slide 19; the express definition is complete and unambiguous. Like the Straight Path and GeneriCo cases, it would be improper to override the express definition in the claim with a description of exemplary embodiments and the specification. And that's precisely what patent owner is asking the Board to do.

Slide 20; because patent owner is importing these three steps, which it admits are just described in the specification, and yet,

patent owner has cited no words of manifest or expressions of manifest exclusion or restriction. And that's what the Federal Circuit has held time and time again is required in order to read limitations from the specification into the claim.

Let's go to Slide 24. Now, patent owner's arguments are also inconsistent with the 138 patent specification. So patent owner argues that when the 138 mentions the virtual machines, it means that a manager has been deployed and is executing to provide specific virtual machines. The specification states otherwise. So as patent owner's expert admits on the right, images that are stored in a repository and then deployed after, and he's talking about this excerpt on the left, where if you look right after this highlight, the images are stored, and then deployment doesn't happen until the next paragraph. And it says after generation of image instances, the control node may deploy the manager to nodes. And yet, if we look at the highlighted text, when images are stored before anything is deployed to create any specific virtual managers on the nodes just while they're stored, the spec is clear that the virtual disk images can be booted on the virtual machines.

So even before deployment and execution of

any managers, the virtual machine images are bootable on the virtual machines. So this contradicts patent owner's supposed logic.

Let's go to Slide 25 now. Patent owner -- I want to talk some about claim 19 and 26. So patent owner has, again, no answer to the fact that the -- all of the independent claims recite that definition for providing autonomic deployment. So patent owner wants to create its own definition, but it has no answer to the fact that a definition has already been provided in the claim itself.

JUDGE DROESCH: Counsel, I have a question. How would you characterize, for example, in claim 19, how would you characterize the claim language to autonomically deploy the image instances of the virtual machine manager on the application nodes? Should that just be given little weight or are we supposed to just consider the by causing clause?

MR. LIANG: Well, Your Honor, I think the by causing clause provides a definition of how that autonomic deployment is satisfied. So it's not that it's given little weight. It's that the claim has this -- explains how that autonomic deployment is done, and so because there is that definition, then when the definition is satisfied, then that limitation is satisfied.

JUDGE DROESCH: Okay. So just by causing the image instances of the virtual machine manager to be copied, you have therefore satisfied the -- to automatically deploy the image instances of the virtual machine manager on the application node. Is that what you're saying?

MR. LIANG: Yes, Your Honor. That's the definition that's recited in the claim.

JUDGE DROESCH: Okay. I just want to make sure I was clear on your arguments. Thank you.

MR. LIANG: Yes, Your Honor. And if you have any other questions, please feel free to interrupt. I'll just keep going, but happy to answer any questions you have.

Slide 26; the challenged claims including Claims 1 and 19, they carefully avoid limitations that require actions to be taken by application nodes. They're drafted from the perspective of the control node and as Your Honors know, it's common to draft claims this way to focus from the perspective of one device to avoid potential divided infringement issues. And this careful partition would be destroyed by patent owner's construction, which requires software to be run at the application nodes.

Slide 27; patent owner requires executing

VMM images as part of deployment, and, therefore, in these last two steps of claim 19, patent owner would require the rules engine to execute virtual machine manager images at application nodes.

Slide 28; but the specification has no mention of the rules engine executing images at application nodes. To the contrary, it actually lists several methods that -- several things that are the rules engine can do. And it says that it can clone, capture, import, export, upgrade software images, but there's no mention of executing images. And, in fact, only the last five columns mentioned virtual machines starting at column 32, line 51. And in those sections, there's no mentions of rules engines, rules engines executing virtual machine managers or anything of that sort.

So the limitations that patent owner wants to import do not even exist in the specification. The patent owner is trying to import limitations out of thin air.

Let's go to Slide 42, and talk about virtual machines. Unless Your Honors have other questions about claim construction.

JUDGE DROESCH: None from me.

JUDGE BEGLEY: None from me.

MR. LIANG: Okay. Well, thank you, Your

Honors. Then I'll move on and begin talking about virtual machines on Slide 42. Oh, and just also my colleague just informed me that no claim construction order has been issued yet, and that there's a claim construction hearing set for February 14th.

All right. So let me talk about virtual machines. Slide 42. We go to petitioner Slide 43, the 138 patent mentions virtual machines as an afterthought. The first 31 columns of the specification describe tiers and other concepts for non-virtualized applications. And virtual machines are only mentioned in the last five columns was just apply those earlier concepts to virtual machines using off-the-shelf software from VMware.

The specification does not identify any purported points of novelty regarding its use of virtual machines or images.

Slide 44; a patent owner's expert admits that virtual machines were known in the art.

Slide 45; the patent owner contends that the 138 patent has somehow a special or different meaning for the term virtual machine.

Slide 46; but patent owner's expert admits

==

JUDGE BEGLEY: Sorry. Just to interject with a question about patent owner's understanding

of virtual machine. It seems that patent owner is ascribing a meaning to virtual machine that's taken from the specification. Specifically, the bottom of column, I think it's 33. Let's see. Bottom of column 32 to column 33. They're tracking that language as what they understand virtual machine to mean.

And that's, "software that creates an environment that emulates a computer platform." Do you agree with that meaning?

MR. LIANG: Well, I don't believe that's what patent owner's arguments are based off of with respect to the virtual machine. And did that language does appear in column 32, but patent owner's never disputed, for example, that Khandekar Figure 1, it shows that there's a software that creates an environment that emulates a computer platform.

So, for example, if we skip ahead to Slide 52, you see Khandekar on the right side and Khandekar shows that there's this virtual machine and it has virtual process, or virtual devices, and things. And so the patent owner never disputed that this is software that creates an environment that emulates a computer platform. It's a virtual computer. And, in fact, we asked patent owner's

expert, well, what's inside a virtual machine. And he said it would have a virtual processor, virtual memory, virtual devices, and a virtual disk. And that's exactly those four things; virtual processor, virtual devices, vertical disc, virtual memory. That's what's taught by Khandekar.

So Khandekar teaches a virtual machine as including, and it satisfies that description in columns 32 and 33 that you just identified.

JUDGE BEGLEY: So do you agree with that meaning, and maybe you can pull up the patent if you would like, but it's 32:67 through 33:2, which patent owner's briefing quotes several times.

MR. LIANG: Yeah. I think we agree that that's how the 138 patent describes virtual machine, and that's how the -- that's what Khandekar teaches, as well. So it, Khandekar, teaches that definition of a virtual machine and as confirmed by its expert here.

JUDGE BEGLEY: Okay. Thank you. That wasn't clear to me.

MR. LIANG: Does that answer your question, Your Honor?

JUDGE BEGLEY: Yes. Thank you. It wasn't clear to me from your briefing how you were reading virtual machine. So I just wanted to clarify that.

Thank you.

MR. LIANG: Okay. Thank you, Your Honor. And I believe patent owner's arguments are not really based on that definition. It's not really clear exactly what they're arguing. You know, they're pointing to what's installed on an image or what's create -- how the images are created, but it really doesn't seem root. Although they quote this language, their arguments don't seem rooted in that language and their expert as shown in Slide 52 admitted that Khandekar in fact teaches these the environment that emulates a computer platform, and it satisfies this definition from column 32.

And, in fact, I think the reason why Khandekar teaches the same virtual machines is evident when you -- if you go to petitioner's Slide 46, it's evident from the fact that the 138 patent does not disclose a new kind of virtual machine manager, and it doesn't describe the internal workings of a virtual machine.

If we go to Slide 47; instead of the 138 patent relied on existing VMware software to provide virtual machines.

Slide 48; and, in fact, the patent, the 138 patent, only discloses one way to provide virtual machines, and that's by using prior art

software from VMware.

Slide 49; Khandekar and Le, the prior art in this case, they're both VMware patents.

Slide 50; and Khandekar does not invent a new kind of virtual machine or VMM either. It just refers to the known use of VMware software. For example, in Khandekar line 7 -- column 7, line 25 through 27 it states that. So then -- and it's also shown here on the left side, which is Khandekar.

So the 138 patent Khandekar and Le, they all use virtual machines that are provided by VMware software. They're the same virtual machines. And that's why when we get to Slide 52, that's why it's no surprise that Khandekar teaches that virtual machine that, you know, including the components that are described by patent owner's expert and satisfying the description that we've just looked at in columns 32 and 33 of the 138 patent.

Slide 53; Khandekar also teaches images of virtual machines, and the 138 patent only discloses one way to create images by using VMware software to capture virtual machine disk image files.

Slide 54; and Khandekar teaches those same virtual machine disk image files as shown in Figures 3 and 4, because it's using that same VMware software to create virtual machine images. A patent

owner's arguments have these arguments that about how Khandekar sometimes refers to images based on the type of virtual machine it contains, so deployable virtual machines like pre-built or custom virtual machines, or instantly deployable virtual machines. But Dr. Shenoy explained, for example, in Exhibit 1029, paragraph 4, that Khandekar's meaning is clear from its context. So, for example, Figures 3 and 4, these show images that the specification explains are stored in a server. So there's stored images.

And, in contrast, if you look at -- if we go back to the Slide 52, Khandekar Figure 1 shows a virtual machine with a virtual processor, virtual memory, et cetera, that's running on a virtual machine manager, and so that is clearly a virtual machine. So Khandekar teaches both virtual machines and images of virtual machines.

Now, if the Board has any questions about virtual machines, I'd be happy to answer them. Otherwise, I would plan to move onto the next topic.

Okay. So I'll plan to move on then.

Slide 55; I want to start talking about the image instances of the VMM, and I want to start with ground 2, because patent owner failed to address this combination, and it invalidates the

challenged claims.

Slide 56; the patent owner response spends almost 20 pages attacking Khandekar alone, and then it spends only a single page on Le, and it also just tries to attack Le alone.

Slide 57; in that single page, patent owner quotes a sentence out of Le, but patent owner never addresses how Le's disk imaging teachings are used in the combination of ground 2 just try to attack the references individually.

Slide 58; as the Federal Circuit has repeatedly held, obviousness cannot be overcome by attacking references individually. Exactly what patent owner is trying to do here. Patent owner chose to avoid the combination of ground 2 in its patent owner response, and it should not be allowed to raise pages of new arguments for the first time on sur-reply or today in this hearing. Patent owner's expert never address the combination and the unrebutted evidence of ground 2 invalidates the challenged claims.

Slide 59; when first introduced -- now, I want to cover how the combination uses ground -- how the combination uses the art. So when first introducing ground 2, Dr. Shenoy explained that Le reinforces Khandekar's teachings with two

implementation details that's mentioned in this top paragraph. The first one is about the ratio of VMMs to VMs, which patent owner never disputed.

And the second one is with disk imaging techniques, and then the next paragraph, it explains how Le teaches you how to deploy images to physical and virtual computers. Le's disk imaging is one of the two stated reasons for the combination.

Slide 60; the petition dedicated an entire section explaining the motivation to combine Khandekar and Le, specifically Khandekar teaches deploying a virtual machine manager along with the virtual machine image. And Le provides implementation details for how that's accomplished using disk imaging techniques.

Slide 61; in the motivation to combine section, Dr. Shenoy explained if you read in the blue highlight that a virtual machine manager is a program running on a physical computer, and then in the yellow highlight, Le teaches techniques for physically imaging the hard disk of a remote computer to deploy a cloned image with an operating system, which would include a Type 1 virtual machine manager and applications, which would include a Type 2 virtual machine manager.

And after this highlight, Dr. Shenoy

explains several reasons why a person of ordinary skill in the art would have used Le's physical imaging techniques to deploy a virtual machine manager (a VMM). So this is the explanation for how and why Le's disk imaging would be used to deploy VMMs.

Patent owner never disputed the motivation to combine Khandekar and Le in this way. And when combined in this fashion, Khandekar and Le render all limitations of the challenged claims obvious.

Slide 62; in the claim analysis sections rely on Le's disk imaging teachings and for limitation 1c, for example, which is the one where the limitation that first recites those images instances of a virtual machine manager.

So the petition and opening declaration, they include an entire section devoted to Ground 2. And they explain how Le's additional disk image teachings compliment and elaborate on Khandekar's teachings, and the blue highlights citing back to that motivation to combine it for how and why Le's disk image teachings would have been combined. And, in particular, how it was obvious to use Le's disk imaging teachings to deploy VMM images.

Slide 63; again for limitation 1f, the petition included an entire section for Ground 2

explaining why it was obvious to use Le's disk imaging teachings to provide the VMMs from the server to the host, and in the next sentence after the highlight, that disk images include the operating system of software applications, which would include Type 1 and Type 2 virtual machine managers, respectively.

Slide 64; when you look at what are these teachings that were cited to show deployment, they show the imaging of physical computers. So on the right side, for example, Figure 5 shows how you write image from a server on the right side to the disk of a destination computer. And then Le describes cloning computers, and there's no dispute that Le teaches old disk imaging that writes to the physical hard disk of a physical computer.

So let's go to Slide 66. This distinction between physical and virtual images was raised in the opening materials and even included in the definition of a person ordinary skill in the art that patent owner's expert agreed with. And that definition includes the knowledge to create images of physical and virtual machines, including images that include VMMs.

Slide 67; now why would a POSITA have been motivated to use Le's teachings to store

an image of a virtual machine manager? Well, for one, because the petition cited two references showing an express teaching, suggestion, and motivation in the art to store images of virtual machine managers. And an express teaching, suggestion, or motivation is a classic motivation under KSR and patent owner argued over and over that we cited no evidence. It would be obvious to store separate images of a virtual machine manager, and yet, here's the evidence, and it was cited in the petition.

Slide 68; now, patent owner contends Ground 2 was a new argument that Le's teaching of imaging wasn't raised -- was raised for the first time on the reply of -- and the support, patent owner contends that Dr. Shenoy could only point to his reply declaration when asked about physical images. Not so. If you read his actual testimony in his deposition as quoted here on Slide 68, Dr. Shenoy pointed to paragraph 58 of his opening declaration, which we just looked at, and Dr. Shenoy explained how in that paragraph of his opening declaration, he talks about how a VMM is a program running on a physical computer, and that Le teaches techniques for physically imaging the hard disk of a remote computer.

That's one of the two reasons why Le is

being combined with Khandekar. So the Ground 2 was presented in the petition. It can't be a new argument.

Slide 69; and patent owner contends that Ground 2 is new, because they failed to address it. And really if you look at their arguments, the patent owner's arguments ignore the fact that the virtual machine manager image is physical. The VMM image is physical while the virtual machine image is virtual. As shown in Slide 69, patent owner argued that Khandekar requires including the VMM within the same image as a VM, but patent owner's expert offers no explanation of how the VMM, which is a program running on a physical computer, could be included in a virtual disk image.

They've offered no evidence that this has ever been done, and it just makes no sense.

Slide 70; now, Khandekar alone also teaches images of a VMM.

So Slide 71; so Khandekar says that the VMM can either be pre-installed on the host or installed along with each VM. So the fact that there's a choice to install these together or separately demonstrates that they're separate images.

Slide 72; and, in fact, Khandekar uses

virtualization. The reason the purpose of using virtual machines to begin with is to make their deployment as Khandekar explains, substantially independent of the overall physical hardware.

Slide 73; and as Dr. Shenoy explained, that you using separate images for the manager and the virtual machines, facilitates this independence that Khandekar seeks, because that one virtual machine image can then be run on any computer where the manager is installed regardless of that physical hardware.

But in contrast, a monolithic image that patent owner suggests would tie the virtual machine to a specific manager on a physical computer and defeat the purpose that Khandekar teaches and the entire purpose of using virtual machines.

Slide 74; the patent owner's expert agreed. The primary reason in the prior art for using virtual machines is to provide independence from underlying hardware.

And if you look at the bottom quote, this allows the same application to run on different physical hardware. So even if it were possible to create a monolithic image with both managers and virtual machines, patent owner's expert never explained, by the way, how or why that would be

done, but even if you could do it, it wouldn't make sense, because it defeats the primary purpose for using virtual machines as patent owner's expert admits here. It would not make sense to interpret Khandekar in a way that defeats its purpose.

Now, I'll just stop my presentation here and ask if Your Honors have any questions.

JUDGE DROESCH: No questions for me.

JUDGE BEGLEY: Same here.

MR. LIANG: Okay. Well, thank you, Your Honors. Then I will reserve the rest of my time for rebuttal.

JUDGE DROESCH: Okay. Thank you.

All right. Counsel for patent owner when you are ready, make your appearance and you may start your presentation when you're ready.

MR. YOUNG: Thank you, Your Honor. This is Dan Young from ADSERO IP representing the patent owner, Avago Technologies.

So if we could start just very briefly on Slide 1. This is a figure that is addressed in the institution decision and in one that's a centerpiece of the briefing between the parties.

And as you can see, this is one example, but this is Figure 20 -- Figure 24 shows -- again,

this is on Slide 1. As a physical node and then a virtual machine manager running on that. Virtual machine manager creates the virtual machine, which as Your Honor noted, that that is our definition taken directly out of the specification in bottom column 32 and relates to this emulated environment, and then you deploy the operating system and the applications on top of that emulated environment.

So if we go to Slide 2, this is just the --

JUDGE BEGLEY: Just to jump in on that. So are you asking for us to adopt that as a construction of virtual machine? Because it's not included in your main construction analysis, but it seems like at least you're referring to that as the meaning throughout and at least one part of your preliminary resp -- sorry -- your patent owner response seems to refer to it as a construction. So are you asking -- are you proposing that as a construction of virtual machine?

MR. YOUNG: Your Honor, as the patent owner did in the District Court litigation, the individual terms, virtual machine manager, we've always applied those in plain and ordinary meaning.

Now, I'm going to get into the order of steps, which those individual terms show up, yes.

Within that construction of the order of steps, then, yes, that is what a virtual machine means in the context of this path.

Hopefully that answers your question.

JUDGE BEGLEY: I'm not sure I understand. If you could just explain that again for me. I'm sorry.

MR. YOUNG: Well, let's go directly to Slide 23, and then this will have the claim language right there.

So as we've discussed and I'll go back, you know, the Federal Circuit as the standard for whether or not -- and this is -- sorry to go back and forth. But on Slide 3, the Federal Circuit has a standard for what -- for how an ordered sequence of steps is whether that is interpreted within a claim. And the Board asks us to address that in their institution decision, both with the Claim 1 and claim 19 and 26.

And what the Federal Circuit says is, the order of steps is either the specification directly or implicitly requires an order of steps, or the claim language, as a matter of logic and grammar, requires the steps to be performed in a written word. And as we've said in our briefing, both of those situations apply.

So if we go back now to --

JUDGE BEGLEY: Just because you've raised this quote, I have another question about that.

So this Federal Circuit case, the Mformation Technologies that's cited here on Slide 3, that case addressed a method claim, right?

MR. YOUNG: That's right, Your Honor. And so if you look at Claim 1. I'm sorry, Your Honor, did you --

JUDGE BEGLEY: Well, sure. Thank you. So most of the claims challenged here are not method claims. So are you asking us to extend the Federal Circuit case law on order of steps regarding method claims to non-method claims?

MR. YOUNG: Well, I'm not sure what needs to be extended, but it certainly is applicable to all the claims in this case, particularly, Claim 1 is a system claim. Claim 19 is a method claim. The parties on both sides have treated all independent claims, Claim 1, 19, and 26, the same. So, again, when looking at the claim language as a matter of logic and grammar, are those requirements there. So what we would say with respect to claim --

JUDGE BEGLEY: But if it -- sorry. You can continue, sorry.

MR. YOUNG: So going back in Slide 23 is

the actual claim language. And I think it's helpful to go through this. There are two big buckets, if you will. There are those -- the software image repository and then there's the control node. Software image repository is the beginning of the claim, control node is the second half of the claim.

In the software image repository, it has two specific things that must be stored there. One is a plurality of images, this is the virtual machine manager; and the second is the plurality of image instances of the plurality of software applications that are executable on the plurality of virtual machines, and the antecedent basis for the plurality of virtual machines is in the first part of the claim where it says that with respect to the virtual machine manager, when executed on application nodes, the image instances of the virtual machine manager provide a plurality of virtual machines.

So going back to your question at the outset about the meaning of the term virtual machine, the virtual machine is the emulated environment in which you deploy the applications onto, and that's the plain language of the claim.

So this is where you define the elements, right, that are going to be deployed, and then when

we -- to answer your second question with respect to the step -- yes, Your Honor.

JUDGE DROESCH: Counsel, I had a question. Now, you're saying that Claim 1 requires actual deployment, but I don't see anything in the claim language that requires actual deployment of the virtual machines. I see that the plurality of image instances of the virtual machine manager but that it's executable and when it's executed. It's not that it's actually executed. Can you elaborate on why it is that is required for those virtual machines to be deployed?

MR. YOUNG: Yes, Your Honor, I will. When that capable language again occurs in the first note I mentioned the first bucket of the claim, the software image repository, and that's because those images are stored there, but they haven't been deployed yet, right. And we'll get to later on, we're talking about the examples in the specification that show how that deployment works. So that's why it would say executable, because they're in the software image repository part of the claim, and that's where they're defining those elements.

In the control node element, which is talks about the deployment itself, it says it has an automation infrastructure to provide autonomic

deployment of the plurality of image instances, and this is to answer your question, Your Honor, earlier you were asking counsel for petitioner about the by causing language, right? That is -- that's the how of the deployment, right. But as the, you know, as the petitioner ignored before that language is the where language, where are we depositing?

We show how we deposit it by causing the copy, but where are we deploying it? Well, the first part of that tells you exactly where. You're deploying the virtual machine manager on the application, so if you look to the right Figure 24 shows virtual machine manager be deployed on the physical node, the application, right? So then -- so that's the first one.

JUDGE DROESCH: Counsel, I have another question. I'm sorry. Counsel, I have another question for you, but how do you explain the language of to provide? It doesn't actually require that it's providing; it's to provide. Can you explain how we're supposed to interpret that?

JUDGE BEGLEY: And just to, to follow on that. I think it would help me, at least, if you could just bring us to the claim language and just focus on the language of Claim 1, not the written description or expert testimony, and just walk

through how the claim language itself requires this three-step process.

So if we could just -- I don't know if that adds on, to me, at least, it seemed to follow on to Judge Droesch's question, but I think I'd like to focus on the claim language, and if we could go to a slide where it's just Claim 1 and you could show us where you're getting the three-step process and walk us through that, I think it would help me.

MR. YOUNG: Yes, Your Honor. And I was before from Slide 23. I apologize if that didn't make that clear, but Slide 23 is where the actual claim language is on the left-hand side of the slide.

So -- and I want to make sure that I'm addressing the question. So the language with respect to the virtual machines were -- and this is again in Claim 1 in the software image repository section of the claim, the upper, the upper part of the claim, the virtual machine managers, and it says, when executed on the application nodes, the software, the image instances of the virtual machine manager provide a plurality of virtual machines.

And so then -- and that's where they're defined. They're defining the software image. You

have two sets of images, you have the images of this virtual machine manager, and you have the images of the plurality of software applications that are executable on the plurality of virtual machines. That's all in the software image repository section of the claim language.

The second part of the claim language, and again, this is on the left-hand side of Slide 23, talks about the control node, and the control node is -- has the automation infrastructure to provide autonomic deployment of the plurality of image of the virtual machine manager on the application.

So that's -- as I mentioned before, that's the where, where are we doing the deployment? The how the deployment is by causing the plurality of images. This is the virtual machine manager to be copied from the software image repository to the applications. And that's -- so that's the first step of the deployment process.

You have to copy the image instances, the virtual machine managers, to the application nodes, and then you provide autonomic deployment of the plurality of image instances of the software applications. And, again, here's the where, the where are you deploying it on the virtual machines? And as you notice that the petitioner, all of this

briefing, they never addressed what the meaning of on the virtual machines mean. It's never addressed. And they're and from our perspective are reading that out of the claims, because that's where the plurality of images as the software applications have to be deployed.

And so, in other words, if those emulated environments are not created, there's no place for the software image, the software, the images of the software applications should be to be deployed onto.

And that as we, as I, you know, in this Slide 3, the claim language reflects exactly what's going on in Figure 24, right? So the virtual manager --

JUDGE DROESCH: Counsel, I guess I'm a little hung up on the claim language. So if we skip over the portion of the control node, portion of Claim 1, and we skip to the, to provide autonomic deployment.

So this is just following on with the control node that comprises an automation infrastructure to provide autonomic deployment of the plurality of image instances of the software application on the virtual machines. It doesn't actually say that you have to deploy it. It's like this -- there's a purpose of providing that

autonomic deployment, and you do that by causing the plurality of image instances to be copied.

I guess I'm just trying to figure out like why there's actual deployment required based on the language in the claim. Can you explain that?

MR. YOUNG: Yes. So the -- in order to go through, so in other words, in order to have the capability to do these things, as an example, the images of the software applications have to be able to be deployed onto the virtual machines. And so the only way the virtual machines exist is because they are created by the images as the virtual machine managers.

The virtual machines where you are deploying the images of this applications, don't exist absent execution on the application nodes. And that's clear from the many examples in the specification that we can go to, but they are highlight in the first half of the slides.

And then to answer your second part of the question is, if you look at claim 19, which is expressly a method claim, there are four steps, the two storing -- and this is again on Slide 29. And in Slide -- and again, the parties had treated all three of the independent claims in this case, Claim 1, 19 and 26, the same, but you have four steps; you

have two storing steps, and then you have two executing steps and then you have two executing steps, and the executing steps mirror exactly the language in Claim 1.

So again, on Slide, 29, we have the claim language, and you have -- the first storing is you store the plurality of images. This is the virtual machine manager. Then you have to find the virtual machines when executed, they provide the virtual machines, then you have another storing limitation with respect to the plurality of images. This is the software applications, and then you have two executing steps. You execute a rules engine to perform. So this doesn't say to provide; it says to perform operations of autonomically deploy the images, this is the virtual machine manager, on the application node. That's the where, where are we deploying it on the application nodes? The how by causing the images, this is the virtual machine manager, to be copied on the application nodes. That's actual deployment.

And then the second execution step says executing the rules engine to perform operations to autonomically deploy the image instances of the software applications. Here's the where on the virtual machines by causing the images, this is the

software applications, to be copied on the application node. So this language in claim 19, which is expressly a method claim, there is exactly the language of Claim 1, which is obviously indisputable of the system claim. And then -- and one issue that I think is important is with respect to the virtual machine language, if you go to Slide 25, and again, this refers back to the specification, but their experts said, the only way that a virtual machine is created is by the execution of a virtual machine manager. And again, that's on the right-hand side of column 25. And there's two examples, Figures 26 and 27, which are examples in the slides in the first part of this presentation that shows exactly how that's done.

So, and I think that might be -- Your Honor, unless you have additional questions about the claim language, I think we can go -- I'd like to go quickly through the specification, because the Federal Circuit standard for determining an ordered sequence of steps, talks about the specification, and it also talks about the claim language, but to summarize the claim language, we're not, you know, the petitioner is -- as we set forth in our sur-reply, the petitioner makes a number of straw man arguments with respect to claim language saying

we don't identify the claim language. We clearly identify the claim language. There's a whole section on our patent owner response dedicated to nothing but the claim language talking about this ordered sequence of steps.

And, and it's, you know, we're not construing individual terms. We're construing the meaning of the claim as a whole based on logic and grammar as the Federal Circuit has instructed us to do. And again, this is in response to the Board's request that the parties brief this as part of the Board's institution decision.

So when we -- so as Dr. Shenoy mentions here on Slide 25, there are two examples of how these virtual machines can be created. Then he refers to Figures 26 and 27. And so those are detailed in the slides starting on Slide 13.

So again, this is the specification, but the specification mirrors the claim language, right? So if you look on the left-hand side, the specification talks about how to do this three-step deployment process, which mirrors exactly what the claim language requires. Step one, you, and again, looking at Figure 24, this is in Slide 13. You create the virtual machine manager, or you, sorry, you deploy the virtual machine manager as an image

instance, it creates the emulated environment, the virtual machine, and then subsequently the image instances of the software application are deployed onto that virtual machine. So that mirrors the claim language.

And again, this is looking at Slide 14. This is the petitioner's expert, Dr. Shenoy, saying exactly that, that the virtual machine manager is executing on the application node or the physical node, and that's what creates the virtual machine. And that's also highlighted again on Slide 15 at the bottom.

It says the virtual machine in the context of the patent, and in this example is software that creates an environment to emulate a computer platform. That emulated environment is what the application nodes are deployed on. And that goes back to the claim language saying where you are deploying it, or deploying the software images, the software applications. You're deploying them on the virtual machines, which is exactly what Figure 24 shows.

Figures 25, 27 and 28, which are also detailed in the Board's institution decision, shows how this -- again, how does this work. First you deploy the virtual machine manager, the virtual

machine manager creates the virtual machine. That's highlighted in the claim with other specification language on the left-hand side of Slide 16. Again, Dr. Shenoy did not disagree with that.

On Slide 17; says once the virtual machine manager --

JUDGE BEGLEY: Just to jump in again with respect to Slide 16. The quote of the specification here for Figure 27 at the very top is talking about it being an "exemplary procedure," and I believe there's similar language with exemplary embodiments and exemplary for the parts of the specification that you're relying on for this three-step process. So can you -- and I believe that your response expressly acknowledges that these are just examples. So can you address how you're being consistent with Federal Circuit case law holding that we should not import exemplary embodiments from the specification into the claims?

MR. YOUNG: Yes, Your Honor. And there's two parts to that answer. The first part, if you go back to Slide 3, we're talking about ordered sequence of steps. And that's where the Federal Circuit says, you look at two different things. You look at the specification, and you look at the claim language. Either one, the specification or the

claim language, can dictate an order and sequence of steps.

JUDGE BEGLEY: Yeah.

MR. YOUNG: So what we're saying, and I mentioned that Dr. Shenoy said there's only two ways that a virtual machine -- only two examples of how a virtual machine is created in the specification. And both of them are with an image instance of a virtual machine manager executing an application node. So the claim language itself requires the deployment on the virtual machine.

So we're not importing any limitations from the specification into the claims. We're saying the claims themselves mirror the only two ways, the only two examples of how virtual machines are created on the application node. So the claim language and the specification align exactly with that. That's why we always, as an example on claim -- on Slide 23, as an example, we match the claim language and we show exactly how the claim language reflects exactly what's in those exemplary embodiments. So we're not reading specific limitations from an embodiment into the claims. We're showing how the claim terms themselves reflect the actual examples in the patent. And those are examples in Figures 25. I'm sorry, Figure --

JUDGE BEGLEY: Are you saying that the specification requires this order of steps or that there's exempt -- there's examples in the specification that have these steps?

MR. YOUNG: In order to deploy the software, the app -- the images of the software applications onto a virtual machine, those steps are required. They're in every example, and the only way you can -- and again, this is where I said at the outset, the petitioner never wants to address. Nowhere in their briefing did they address on the virtual machines. That's actual claim language. Again, we're talking about Slide 23, the green highlighting, if that shows up on your slide, but that's at the bottom. It says, "the software applications on the virtual machines." Those have to exist, and the only way they can exist and the only way -- and the petitioners, we just talked about in a previous slides, the only way the specification talks about the virtual machines existing is through the execution of a virtual machine manager on an application node. It's the only way it happens.

So you can't -- and looking at again, Slide 24, I'm sorry, Slide 23, Figure 24 the virtual machine manager creates that virtual machine. That machine is the foundation is the emulated

environment on which you deploy the software applications. If, unless that exists, there's nowhere to deploy.

And this -- and I can move on to the prior art, but this is the primary distinction between the prior art and this patent. This patent talks about this three-stage process, and there are advantages of doing it this way. And I can skip ahead just for a moment, then I'll come back because this is important.

This shows that there is a very big distinction. You look on Slide 55, this gives you an example. If on the left-hand side you have the image instances of the virtual machines, right? So we'll look at those and let's say it's A, B, C, D, that's on the left-hand side, and then you have image instances of the software applications, right?

And so if you deploy them in these different steps, the advantage of that is if, for example, you want to change an image instance, you want to change the format, you want to add functionality or whatever, you only have to change one image, right? But in the prior art, the way Khandekar Le does it, they talk about this VM, which we can go into that. That's a different virtual machine in our patent, is this emulated environment,

a VM in Khandekar and Le is an entire disk file, and that's what you see on the right-hand side of Slide 55.

You have all these different images, so these are all one image, a single image, a single deployment step. And if you wanted to change, you know, part B as an example, you have to change each one of those image instances individually, you have to go in and change everything. And this is only an example of, you know, of five image instances and five software, because that's all we could put on the slides. If you had hundreds of these, you could see that if you wanted to make one change, you have to go in there and change every image instance. You couldn't just change the one. And that's why the advantage of our patent, the Sundar patent, is you have these images stored separately in a software repository, and then you can combine them as you need. But if you need to change one or update one, you only have to change one. You don't have to go back in and change all of them as the combined image.

So going back then to the claim language on Slide 23, that shows how this, you know, the virtual machine going down first, creating the environment and then deploying the software, the

image instances of software applications on that environment, that captures that advantage of doing it that way.

And so to summarize, plain language reflects exactly how the specification teaches these deployment steps.

So, Your Honor, as you noted correctly in the example I was reading on Figure 27, it does absolutely say, and I'm looking on Slide 16, is it absolutely says an exemplary procedure, but the exemplary procedure with respect to these steps is actually what's reflected in the claim language. So the language of the specification (inaudible).

So I want to -- unless Your Honor has additional questions about the claim language itself, I'd like to discuss the prior art.

JUDGE BEGLEY: I do have one last question, and it goes back to the Mformation case cited on Claim 3. I'm sorry, on Slide 3, about the ordering of steps. And I'm just still having trouble understanding how we're getting from steps, and ordering of steps in method claims, and using that in system claims?

MR. YOUNG: Well, Your Honor, I think the best way to address that I think is to look at Slide 29.

So Slide 29 deals with claim 19. Claim 19 is a method claim expressly, and in Claim 19, it has four steps: two storing steps and two execution steps. And so this language talks exactly how those steps are reflected, and I can go through it again, but those are the steps that are required in this method. And they're actual storing steps and actual execution for the execution of the rules engine to do the actual deployment, right? And so -- you claim 19, if you compare -- I think the best way to answer your question is compare claim 19 to Claim 1, and Claim 1 is on page Slide 23, and Claim 19 is on Slide 29. So if you compare those two, both claims have storing requirements where a plurality of images of the virtual machine manager and a plurality of images instances of the software applications reside. The both of the claims say that the plurality of images of the virtual machine manager provide the virtual machines. Both the claims say that. Both the claims have a deployment process, which requires copying these two sets of images to the application nodes, and both of the claims require that the first time the deployment happens, you're deploying the image instances, the virtual machine manager, on the application nodes. The virtual machine managers provide the virtual

machines, and then both claims have a step that the image instances of the virtual software applications are deployed on the virtual machines.

The virtual machines are created by the virtual machine manager. So to answer your -- to summarize your -- the claim language of Claim 19, which is clearly a method. There's no question falls directly within the claim language of the Mformation case on Slide 2 -- 3. That's -- there's no question there. And if you look at the claim language of Claim 1, it has all those same elements.

So does that answer your question, Your Honor?

JUDGE BEGLEY: I think so. Yes. Thank you. I just -- I'm still not seeing how we're talking about steps in non-method claims, but I think you've explained your position.

MR. YOUNG: Well, I mean, I think we've maybe the better way to say it is with respect to Claim 1, which is a system claim, there is required functionality to deploy the images with respect to those steps. So the functionality, that capability, has to be there to do the deployment in those three steps. And that's exactly what the prior art does not disclose. It does not disclose art that has the capability of doing these three steps and certainly

no disclosure of that.

So if you -- the steps are expressly required in Claim 19, which is a method and the -- what's expressed in Claim 1 is the functionality has to be there to perform these three steps.

And, again, as the parties have -- as we've addressed in the briefing and in the slides, the parties have treated all three claims essentially in the same manner, because the language mirrors each other in expressly; both one is a system, one is a method, and then Claim 26, of course, is on Slide 31, and that's a non-transitory computer readable medium.

And so the functionality needs to be there for the steps for Claim 1; the method, the steps are expressly required and expressly set forth, is a better way of saying it, in Claim 19, and those steps are expressly required through the functionality in being claimed.

So if we go to the prior art that's starting on Slide 33, this will, I think, highlights the difference in --

JUDGE DROESCH: Counsel, I wanted to interrupt you for a moment, and let you know that you're about half an hour into your one-hour time. I didn't know if you wanted to reserve any time for

rebuttal.

MR. YOUNG: Oh, yes, Your Honor. I'll reserve 15 minutes, so I'll go 45 minutes on this part and reserve or reserve 15.

JUDGE DROESCH: Okay, great.

MR. YOUNG: Thank you, Your Honor.

Okay. So on Slide 33, it's -- we're talking -- oh, before I get there, I skipped one thing. If you can go back one slide, Slide 32, this I think also shows you the difference between the parties.

The petitioner is asking for a claim construction, and this on the left-hand side of Slide 32, and the claim construction that they want is that for the term a plurality of image instances of a plurality of software applications that are executable on a plurality of virtual machines. And what they say is that, well, this should be satisfied by dependent claim 2, and dependent claim 2 just simply says that wherein the image instances of the plurality of software applications comprise virtual disk images.

And there's no question, there's no dispute that the images of a software application can comprise virtual machines, or, I'm sorry, virtual disk image files. There's no dispute about

that, but that's not the totality of that claim, claim term. That claim term says that are executable on a plurality of virtual machines. And this goes back to my point before. The petitioner never wants to address the term on the virtual machines. And so what they want you to do is to say, okay, these are virtual disk image files, virtual disk image files are the software applications, and, therefore, it doesn't matter where you deploy them. That's again goes back to the how, like how you deploy them, but not the where, where do you deploy them?

They want to read out the where you deploy them from their claim construction. I think this point, this particular example, makes that point very clear.

So now going back now to the prior art, and this is -- when you understand the prior art, you understand why they want to read on the plurality of virtual machines out of the plain language, because what Khandekar teaches and the, primary, the main, looking at Slide 33, again, the highlighted sections, the main feature of this invention is this provisioning server, where the user can pick any kind of VMs. The term VM goes back to the, goes back to the, you know, the meaning

of what is a VM and in the -- I think the best way to look at that is if you look at Slide 40 as an example, this provides a good example.

Slide 40 on the right has the VM there. The VM, what does that include? Well, it includes a number of things, but one of the things that it includes is applications. You can see that in element 360.

So what a VM is in the Khandekar and Le is the same, it's the totality of the disk image. And the reason why it's a single disk image is because it teaches. It teaches one single image is being deployed, and they have a number of examples of why they believe that that's an efficient way of doing or a desirable way of doing it. The point is that it is in a single image, and that goes back to when we were mentioning before, there's a distinct advantages in the Sundar patent, which the petitioner never addresses, which show why you're doing it in the two-step between the virtual -- the virtual image instances of the virtual machine manager, and then the image instances of the software (inaudible) doing it in a two-step process, provides a lot of additional functionality and ability to upgrade the applicant -- the different images without having to address every single image

instance like you would do in how Khandekar and Le teaches it.

Looking at Slide 34, this talks about, and this highlights, this distinction between the meaning of the term "machine". In our patent, it's the emulated environment. That's in the top part of slides. And then in Khandekar, it teaches this model virtual machine is a virtual machine that contains certain configuration of a guest operating system, service pack, and applications that have been tested and known to work correctly.

So, in other words, Khandekar, when he talks about a virtual machine is the plurality of the virtual machine. Where our, the 138 patent talks about a virtual machine. That's the emulated environment. That's the platform in which the images instances of the software applications are deployed onto.

And look on Slide 35, Dr. Shenoy agreed with that. So -- and again, this highlights the difference between the M in Khandekar and Le in a virtual machine in the 138 patent. So in the left-hand side is Shenoy testimony, you got the three different types of VMs that are disclosed in Khandekar. And in all three of those, they include an operating system, and instances, and

applications, and that's the left-hand side, that quoted language.

Now, in the custom version, you can deploy it and then install some applications later on, but installation is a different way of deployment and (inaudible). As you remember the claim language, very going back very quickly to Figure 23, doesn't talk about deployment process. It talks about deployment through a copying of an image instance. Those are very different, and the reason they're different is because if you deploy an image instance of an application, it can already be configured. It can be exactly the -- you know, to use a simple example, like in words, you know, what font you want to use, what, you know, font size you want to use, whether you want a portrait, a landscape. You can -- all of those are configurable so that when you take an image instance of that application, or you deploy it, those configurations are already there.

In the example in Khandekar when it talks about installing applications later, that's the actual installation process. You've got to go through the whole configuration after that. So that's much different than what the claim language reflects. And this whole distinction between VM, and Khandekar, and Le, and virtual machine in the 138

patent is expressed acknowledged by petitioner's expert, Dr. Shenoy.

Looking at Slide 36, he says exactly that. He says, "I've also explained that Khandekar sometimes uses the shorthand." When he says VM, he's referring to VM images, and depending times VM means a virtual machine itself.

So it's very important when the Board is looking at Khandekar and Le, that we are looking at the VMs and the instantly deployable, and the custom ones, all of the different three that Khandekar talks about. That VM is much more akin to the image instance of the software applications than it is to the virtual machine emulated environment upon which those software applications are deployed on.

And so going to Slide 37, when it talks about virtual machine manager in Khandekar, it gives very little information as to how the virtual machine monitor is deployed. It has two different options in the left -- the right-hand side of Figure from, sorry, on Slide 37, it talks about that. You can pre-install it; and again, as I mentioned earlier, installation is not the same as copy image instance. It's very, very different. You can pre-install it, so it's already there so that when you deploy the images of the VM in Khandekar, it's

already set up. And then it also says you could deploy it along with, and on the right -- on the left-hand side, it says "included and installed together with the VM."

So if you're installing them together in Khandekar doesn't provide any explanations of how this would work. You know, the petitioner wants to say, well, we're coming up with this argument. It doesn't make any sense. What we're doing is taking Khandekar at face value, because face value is you're deploying it together or you pre-install it. And if you deploy them together, then you're not doing this three-step process. You're not deploying the virtual machine manager first, creating the virtual machine, and deploying the image instances, the software applications on virtual machine. That's expressly not done by Khandekar in either example, the pre-installation process nor when you deploy them together.

And if you go to Slide 38, this deployment is expressly in Khandekar as a single step. The right-hand side. Deployment is the step of taking a staged VM. And again, that's an image instance, the application and everything, the whole image, which is what's shown in Figure 1 and Figure 2 of Khandekar, and you're copying it to the host

machine, a single-step process. Now, that's more efficient in the deployment process, but it doesn't provide the function. It doesn't provide the expandability and flexibility of the process that's described in the 138 patent or even deployed in series, you have more flexibility, as I mentioned on Slide 55. And again on Slide 39, it also talks about these steps.

So if we look now at Khandekar itself with respect to looking at Slide 40, I think this shows up pretty clear. So if you look at what they do with respect to limitation 1d, they point you to Figure 1, which is a prior art. It's not part of the Khandekar teaching, it's prior art, and that shows the VM, which is again, that's the entire image instance and the virtual machine of VMM down below.

It doesn't talk about how those get there. They just show, right? And there's no dispute that a virtual machine manager and virtual -- a VM as described in Khandekar, that was known in the art. The advantage of the 138 patent is, again, how you deploy the special manager, create that flexibility, which we discussed in Slide 55.

Then what they do, and I'll skip ahead a little bit, because I want to get to the Khandekar

-- the example, as well.

If you look in Slide 42, what they do with respect to 1e is they point to a different figure, Figure 2. This is what Khandekar teaches, and if you see in Khandekar, and if you look down at the bottom, it will have the deployment of the VM and the host. They don't even describe the VMM at all. It's not mentioned in Figure 2.

So, and again, what they're doing is if you don't -- and I'm going to skip down to Slide 46. In an antecedent basis obviously the Board's familiar with that concept, and where you initially define something, you have to refer to the same thing.

And as Dr. Shenoy mentions in this testimony on Slide 40, he doesn't know what antecedent basis is, and he provides no opinion that the virtual machines that are first defined in this software repository bucket of the claim, whether that's the same virtual machine that you have to deploy the soft image instances of the software applications onto in the control part of the claim. He expressly says, I provide no opinion on this issue.

And when you do that, then on Slide 47 with Dr. Rosenblum, who is the patent owner's expert and the chairman of the Department of Computer

Science, George Mason University. He says, when you do that, you are ignoring the structure of the overall claim. And then you can just mix and match. You just find words in the various prior art references, ignore the overall teaching. You just find some words in Figure 1, which is prior art. You find some words in Figure 2, which is what Khandekar teaches, and you kind of mix and match because you don't -- you're not applying the antecedent basis to your analysis.

So let's skip ahead. So this whole issue -- I'm going to Slide 58. This whole issue of, you know, what they say, we don't address Ground 2, Khandekar in view of Le. That is incorrect. We talk about in our patent owner response exactly why. Basically Le is supplements the teachings of Khandekar, but it's still at the end of the day, Khandekar and Le both teach. You are taking a single image of whatever's on the computer. There is no disclosure of taking two sets of images, and putting them in the software repository, and deploying them in series. And the references that they talk about while there's some evidence that there are prior art that talks about image instances of a virtual machine manager. That's Exhibit 1014 and 1016. Those are not part of claim -- Ground 2

or Ground 1. And there's no disclosure in those references that those images are even stored in the software image repository. And so what the petitioner did in his reply is they changed their underlying position with respect to Ground 2.

They said, well, we're not just talking about the supplementing that the teachings of Khandekar, what we're doing in this new argument is we're going to say, you take Khandekar, you create an image of a virtual machine manager, then you take -- and then you deploy that, and then you reboot the machine, and then you deploy the image instances of the VMs from Khandekar on top of that. Well, that's absolutely not discussed in the petition and this language with respect to Dr. Shenoy's deposition testimony on Slide 58, there's, you know, seven pages of testimony, which we quote in our reply. And while there are bits and pieces of this argument scattered through the petition, there is absolutely no disclosure of this three-step process using Khandekar and Le that's disclosed or relied upon in the petition. And when I asked Dr. Shenoy repeatedly at the end of that testimony, that's where he said, well, it's in my reply declaration.

So we're not saying that there aren't bits and pieces of this argument in the petition, but

this combination of using Khandekar and Le in this kind of series is not disclosed anywhere in the petition, and that's why we say on the right-hand side of Slide 58, the term physical image and booting the host computer are not found anywhere in the petition.

So with that, Your Honor, unless there are any questions, I'll reserve the rest of the time for rebuttal.

JUDGE DROESCH: Thank you.

I don't believe there's any questions. So if counsel for petitioner would like to begin their rebuttal, when they're ready.

MR. LIANG: Your Honor, before I begin, I would like to just raise an objection to patent owner's new arguments. As we noted in that call on the November 23rd with Your Honors, the Board, patent owner's arguments about these disk imaging techniques, including the contents of physical images, those are all new arguments. They weren't raised in the patent owner response, which only spent a single page and never, you know, trying to attack Le and then Khandekar alone. They never addressed how Le's disk imaging techniques would be used for the combination. And so all the arguments that patent owner raised today about the combination and

about Le's disk imaging teachings, those are new and should not be considered and same with the sur-reply arguments and the scheduling order paper 10, the Board cautioned that patent owner is cautioned that any arguments not raised in the response may be deemed waived, and the Board has consistently excluded those types of new arguments.

If Your Honors have no questions, I'd like to begin, then.

JUDGE DROESCH: Go ahead.

MR. LIANG: Okay. Now, I want to just start with claim construction and go to patent owner's Slide 3, which is PDF page 7 of their presentation.

Now, and I think Your Honor, Judge Begley pointed out a critical flaw in their argument that their entire construction relies on this Mformation Techs case, which is in that case, there is a method claim that recited steps, and that case just dealt with the order of the steps that are already recited in the claim. The steps that exist in the claim as written. It did not deal with the importation of new steps into a claim. And so while Claim 19 is a method, it still does not include the steps that patent owner wants to import. So Mformation Techs is not inapposite to this -- to what we're talking

about here. Instead, we're talking about incorporating three new steps. As we discussed in the opening, we cited in our briefing, the bar, the standard for importing new limitations is very high. There has to be words of manifest exclusion or restriction, and patent owner has not even attempted to satisfy that high standard.

So before we talk about the ordered sequence or the sequence of steps that should be performed, we have to talk about whether those steps should even be brought into the claims, and they have not even attempted to require -- to meet the standard that the Federal Circuit says is required.

Let's go to Patent Owner Slide 28, which is PDF page 32. This process, this case that's cited here, Process Control Corp. It's just about antecedent basis, and as we explained in our opening, antecedent basis contradicts what they're construction, because there's no dispute that this first mention of a plurality of virtual machines that introduces a plurality of virtual machines that the VMM images are capable of providing when executed, and because they share an antecedent basis, then all these subsequent references to the virtual machines, they just simply refer back to the virtual machines that the VMM images are capable of

providing.

We did not ignore the virtual plurality of virtual machines. They said it over and over that we ignored the plurality of virtual machines, the virtual machines, but we explained it that the virtual machines just refers back to the antecedent basis of the virtual machines the VMM images are capable of providing.

We didn't ignore the language, but what patent owner ignored, and their presentation ignores as Your Honor, Judge Droesch pointed out, their presentation ignores the language executable. They ignore that executable language, and I'd also point out if we could go to Slide 32, Patent Owner's Slide 32. That, you know, this construction that we propose it's really just about, well, what are these image instances of software applications. Patent owner didn't dispute that virtual machine disk images satisfy that aspect of the limitation. And that's what this -- that's why it says, you know, it's satisfied, but not limited to a plurality of virtual machine disk image file. It's just connecting the idea of virtual machine disk image files with that plurality of software applications.

And there is also no dispute that virtual machine disk image files are executable on the

virtual machines that the VMM is capable of providing. So everything is entirely consistent.

Now, patent owner argued repeatedly that Dr. Shenoy conceded their claim construction, and they keep citing to his testimony, but if you actually look at all the cites that they have, in each of them, he's just talking about exemplary embodiments and the specification. He's talking about particular parts of the specification, not the claims. And regarding the construction, we also explained how the combination of Khandekar and Le teaches patent owner's construction. This is in the briefing and it's also in our presentation, but I'm going to leave it for the briefing unless Your Honors want additional explanation about that.

JUDGE DROESCH: I don't believe --

MR. LIANG: Okay. Let's just talk about --

JUDGE DROESCH: I was just saying that I don't believe that there's any questions from the panel.

MR. LIANG: Okay. Thank you, Your Honor. let me talk about virtual machines then.

And so Khandekar is focused on how images are deployed to host. It doesn't invent a new kind of VMM, and Khandekar, column 7, lines 25 to 27,

teaches that the general features of VMMs are known in the art and therefore not discussed in detail here. And so therefore Khandekar uses existing VMware software, including from VMware or it just existing VMware software, and it teaches that in column 13, starting line 17.

So if we go to petitioner Slide 79, so this is petitioner's Slide 79. We go to petitioner's Slide 79. So Khandekar, Figure 1 shows the operation of known VMM software. So Khandekar says, hey, just use VMM software, like from VMware. And that's what Figure 1 shows just the operation of known VMM software that existed in the art and how it manages the execution of virtual machines.

As patent owner argue, well, this is prior art, but if we go to Slide 82 -- well, first of all, that's because Khandekar teaches the use of prior art, the prior art VMware software. And if you go to Slide 82, patent owner's expert admitted, he agreed that Khandekar's teachings of the background art are part of the teachings of the patent. And that's clear if we go back up to Slide 50, you can see the teachings in Khandekar and Le of using existing VMware software and the Slide 51, petitioner's Slide 51, explains the motives -- Dr. Shenoy explained on the left, explained many

motivations about why a POSITA would have followed the teachings and use VMware software to provide virtual machines. And on the right side, patent owner's expert admitted that a POSITA would have been able to use VMware software.

I want to point out that regarding virtual machines, patent owner -- if we could go to Slide 52 now.

Patent owner has never proposed a construction that would preclude the virtual machines that are taught by Khandekar, Figure 1, right? Their expert admitted that in terms of that definition, right, that creates a software that creates an environment that emulates a computer platform, that that's what this, he admitted. Well, that virtual machine would have a virtual processor, memory, virtual devices, virtual disk. Those are taught by Khandekar.

Now, that definition does not preclude running an operating system or applications on the virtual machine, which would make no sense because if you precluded running an operating system applications on their virtual machine, it wouldn't do nothing.

So that the definition in the 138 patent does not preclude also running an operating system

or applications. And if this is a comprising claims, so once the Khandekar teaches the limitations of the claim or the definition in the specification, then it doesn't matter if it also teaches additional other things. And patent owner has not precluded proposed any slide or any construction that would preclude this virtual, this virtual machine in Khandekar as shown in Slide 52.

Now, I want to go back to patent owner's Slide 35, which I believe is Slide 39 in their presentation. And patent owner argued that these three, that the patent argued that the pre-built model VM, custom-built VM, and instantly provisioned VMs that these are not the virtual machines. But if you actually -- if you look at the petition, these are the examples of virtual machine images that are -- these are examples of virtual machine images that were relied on by the petition. So if you recall, if we go to Slide 54, then you see a sample of these deployable VMs and instantly deploy VMs, we'll be able to -- sorry, petitioner's Slide 54. If we go to Petitioner's Slide 54, you see that these are the examples of those deployable VMs and instantly deploy VMs. That Khandekar does use shorthand sometimes referred to these images as VMs, but it's clear from the context that these are images, and

Khandekar explains that they're stored on a server. So these are images of a virtual machine, not the virtual machine itself. The virtual machine itself is taught, for example, in Khandekar Figure 1, and they have proposed no construction that would exclude Khandekar Figure 1 or their virtual machine taught inside it.

Now, I'd like to then -- and I'd like to point out also that the model VMs and the fact that these, the fact that some, the virtual machine images have operating systems and applications, that's actually entirely consistent with what patent owner contends is that two-stage process as described for creating images in the 138 patent.

So they --Khandekar and the 138 patent actually use a similar process for creating images where you install the OS and applications, and then you save that virtual machine disk image file. So the fact that these images have operating systems and applications is entirely consistent with -- is inconsistent with the patent.

Do Your Honors have any questions about virtual machines? I'd be happy to answer any questions you have.

JUDGE DROESCH: I don't have any at this point. I believe the panel does not either.

MR. LIANG: Okay, thank you, Your Honor. Well, then in that case, I'd like to go to patent owner Slide 51, which I believe is PDF page 55 and point out -- and I'll talk about some of the separate images of -- for the virtual machine manager.

Now, the petition cited these two references to as express teaching, suggestion, and motivation for why you would make the combination, right? That's one of the classic motivations under KSR that's cited. And so these are just teaching suggestion and motivation in the prior art, and notably patent owner never disputed the reason why they're used. Both of the references, which are French and Powell, both of them were cited in the petition to show that teaching, suggestion, or motivation for having separate images of virtual machine managers, separate images of a virtual machine manager, and patent owner did not dispute that these references in fact provide that express motivation; instead, all he argued when he argued today and all that's written in Dr. Rosenblum's declaration is a dispute about whether these also teach storing things in a repository, but for Ground 2, the petition cited Khandekar and Le's teachings about the software repository, including Le's

Figures 5 and 6, which show a plurality of physical images being stored at the server. So they did not dispute the reason why these two references were used.

If we could go to the patent owner's Slide 56, which is PDF page 60. Or actually let's go briefly then to Slide 55. Let's go back one slide, Slide 55. I'll point out that patent owner's expert admitted that the prior art used virtual machines to provide independence from underlying hardware before the 138 patent, and that's the primary reason they were used in the prior art.

So that kind of independence is actually what's shown on the left. Patent owner's notion that, hey, the 138 patent invented a new kind of virtual machine or a new use of virtual machine images is contradicted by its expert who admitted that the thing on the left was the primary reason virtual machines were used in the art to begin with. Because it allows that same software application to run on different underlying physical hardware platforms.

A monolithic image as shown on the right is something that patent owner -- it undermines the very reason why virtual machines are used. Now, patent owner wants to twist Khandekar into saying

that somehow the prior art uses this monolithic image, but it actually hasn't provided any evidence that this was ever done in the art. And it doesn't make sense to twist Khandekar into using some kind of an image that was never used. It just doesn't make sense. It defeats Khandekar's stated purpose of providing independence for the deployment of its virtual machines. It just makes no sense.

Now, I'd like to go to patent owner Slide 56 to the next page or the next slide, patent owner 56. Now, this -- if you look at that second red text, it says, and patent owner admits that Le discloses whole disk imaging with virtual machine managers software already installed. So that's why the petition used Le for in Ground 2, to write a disk image with a virtual machine manager software already installed onto a physical computer, right?

So even though it, patent owner, didn't address the use of disk imaging in the patent owner response, it's now conceding that Le discloses image instances of a virtual machine manager. But when talking about Le, patent owner makes three critical errors, and I'm going to go through each of these three.

First, patent owner conflates applications that run on a physical computer with virtual machine

applications. So if we go to patent -- so this is illustrated. And let me go back to petitioner's Slide 61. So let's go back to petitioner's Slide 61.

So the virtual machine manager, the VMM, is a program running on a physical computer. It's either the OS or the application running on a plain old physical computer.

Go to Slide 64. And there's no dispute that Le teaches whole disk imaging that clones the hard disk of a physical computer. And if we go to Slide 65, so Le's disk image that writes the OS and applications that run on the physical computer, which here would be the virtual machine manager. The VMM is the program that runs on the physical computer. And when the VMs runs, it manages virtual machines and applications can run on the virtual machine. So for the 138 patent, the image instances of software applications are virtual machine disk images. And that's admitted by the patent owner in patent owner response, page 43.

So if we go to Slide 54, petitioner Slide 54, the Khandekar is what teaches those virtual disk images, right? Khandekar teaches virtual disk images. Le is using the combination to teach physical images. So Khandekar's virtual disk images

satisfy the claim limitation for image instances of software applications. In fact, if you go back to patent owner's Slide 56, or if you go to Slide 57, patent owner Slide 57, which is PDF page 61, patent owner and its expert admit that Le's virtual machines, it's the virtual machines that correspond to the claimed software application image instance.

The quoted part of Le on the left is talking about a different kind of application. That quoted part of Le on the left is talking about cloning a physical computer. And when it refers to an operating system and applications on the cloned computer, Le's referring to programs that run on a physical computer.

So the programs that run on a physical computer, the thing that's cloned would be for the combination of Ground 2, the virtual machine manager.

So the second critical point is that patent owner's misrepresenting Le, because patent owner wants you to believe that Le's physical image somehow includes virtual machines.

And, of course, none of this was argued in the patent owner response, because they never addressed how these images would be used in this combination. But now they want you to believe that

Le's physical image somehow includes virtual machines, but on the right, I want to point out that this is a misquote here.

Like when you look on the right side, this highlighted text quotes the cloned virtual machine. But if you look at the cited text on the left, all it says is "cloned computer." Patent owner injects virtual machine into this quotation when in fact the cited text, this entire paragraph, there's no mention of virtual machines, and that's because it's talking about physical computers.

And as we walked through earlier, Le's teachings or disclosure of whole disk imaging was applied to physical disks to install the virtual machine manager, not the virtual machines. It's not about the virtual machine applications.

And patent owner is trying to take this quote from Le and turn it into a discussion of virtual machines when it says nothing of the sort.

Now, a third critical error is that patent owner overlooks the fact -- and if you look at the highlighted text on the left here on Slide 57, Le says a set of core applications. And if you -- so Le here is teaching that you can clone a computer with a complete software stack consisting of an application and set of core applications.

Now, the next important sentence is important. The resulting cloned computer, it can then be further customized and personalized using the customization techniques described earlier. Well, what are those customization techniques? Those are explained in Le Exhibit 1005, column 49, line 56 to 63, where Le states customization can also be used to install additional software. It is common to configure a template image with a core set of applications that most clones are expected to need. Non-core applications, that is, those that might be needed by some clones and not others, are generally not placed in the template image in order to minimize the images' size. Instead, they can be selectively installed on a per-clone basis using customization.

So patent owner wants you to believe that Le's image somehow has everything a host computer will ever run based solely on dimension of core applications, but Le is clear that core applications are only those used by all clones, which for Ground 2 would be the virtual machine manager, because that's that core program that's needed for all hosts in order to run the virtual machines. The core applications would not include virtual machines, because both experts and the prior art all teach

that the whole purpose of using virtual machines was to allow different virtual machines to be deployed on different hosts.

And we explained this in petitioner's Slide 72 to 74, which I talked about earlier. So the whole reason to use virtual machines is to allow these different machines to be run on different hosts independent of underlying hardware. Patent owner ignored that distinction between physical and virtual images. Its expert never addressed whether it's even possible to create a monolithic that's somehow both physical and virtual, and it just makes no sense.

So patent owner's arguments for claim construction are contrary to their District Court position.

So I just want to close by going back and hitting on these and summarizing these points, right?

For claim construction, their arguments are contrary to their District Court briefing, which notably the patent owner did not even address in its presentation, has no answer for the fact that in its presentation, it said the virtual machine, in District Court, it said the virtual machine manager just has to have capability. It doesn't actually

have to be running and managing virtual machines. It has had no answer for that in its presentation today. So it's claim construction is contrary to their District Court briefing. It's contrary to the express definition of deployment in the claims, which again, they had no answer to. And it's contrary to controlling Federal Circuit precedent, which they never even tried to satisfy that in order to import a limitation from exemplary embodiments into the claim or embodiments into the claim, there have to be words of manifests restriction and exclusion, and they never even tried to point that out in their briefing.

As for these other arguments, patent owner fail -- patent owner's arguments about virtual machines are incorrect, because Claim 1 is a comprising claim. So it doesn't matter that Khandekar teaches an operating system and applications being installed on the virtual machine. And, in fact, the 138 patent teaches that, as well, because that's part of the image creation process. That's how you create the images, you install operating system and applications, and you save an image.

So there's no contradiction there at all. And even based on patent owner's two-phase creation

process, Khandekar is entirely consistent with the 138 patent.

So it's a comprising claim. There's nothing that precludes -- they've never proposed the construction that would exclude the virtual machine of Khandekar Figure 1 and their patent. And the fact that it's prior art is something that while they've argued, their expert has conceded that those prior art teachings are part of Khandekar's teachings, because Khandekar says to use no virtual machine manager software, including from VMware.

So the virtual machines taught by Khandekar, and Le, and the 138 patent, they all use -- they're all provided by that same software. And the virtual machine is taught by the prior art.

And with respect to the image instances of the software -- the VMM, patent owner failed to address the combination of Ground 2, they failed to address Le's disk imaging teachings, which clearly teach the image instances of the virtual machine manager.

And, in the end, if you step back, patent owner is trying to use VMware software to patent something that VMware already did in the prior art. And it makes no sense.

I'd like to just pause and ask the Board

if it has any questions; happy to answer any questions.

JUDGE BEGLEY: None from me. Thank you.

JUDGE DROESCH: I don't believe the panel has any questions.

JUDGE ENGELS: No questions.

MR. LIANG: Well, thank you, Your Honors.

JUDGE DROESCH: All right. So that concludes your presentation, counselor?

MR. LIANG: Yes, Your Honor.

JUDGE DROESCH: Okay, great. Thank you. Thank you very much.

Counsel for patent owner, if you'd like to begin in your rebuttal.

MR. YOUNG: Yes. Thank you, Your Honor. I'll try to go through these in order. There's a number of issues. I'll try to go quickly, but not, not, not too quickly.

The first with respect to their objection about this new argument. I think that's all laid out very clearly on pages 23, I'm sorry, 21 through 23 of our sur-reply brief where we have many, many citations to the patent owner response with respect to those arguments. Again, the big picture is this argument that they claim now that you can use the -- do one deployment, reboot the machine, and then do

use Khandekar with a second deployment. That's a new argument. That's why it wasn't rebutted in the patent owner response, because it was never argued in the petition. And that's very clear on page 21 of our, of our sur-reply.

With respect to the Mformation Techs case, we're not importing steps into the claim language as we went through in my initial presentation. Claim 1, Claim 19, and claim 26 expressly provide for those individual steps. They have all three of those claims out of the software image repository, where two sets of images are contained, and all three of those claims have some type of control node or other mechanism to deploy the image instance of the virtual machine manager, create a virtual machine that software applicate -- the images is a software application deployed onto. So we're not importing anything from the specification, not importing any steps into the claims. We're doing exactly what the Federal Circuit requires in on Slide 3 of our presentation, which is the claim requires an order, ordering of steps when one specification directly or implicitly requires in order steps. In our briefing, we extensively set forth exactly why -- how the specification does that. And there's no different examples. They're

are all done exactly the same way. Every single example in the specification does it that way.

And then the federal court case goes on to say, where a claim language, as a matter of logic or grammar requires the steps (inaudible) a written order. And this goes back, you know, that with respect to Slide 28 of our presentation, you know, petitioner's counsel says. Well, you know, we're, we didn't ignore antecedent basis. And we talked about the virtual machines, and that's -- and we didn't ignore that. They didn't ignore the virtual machines, what they ignored was on the virtual machines. And as you see in the claim language, that's where you were deploying it on the virtual machines, and that's on Slide 28, the bottom clause of the claim. That's where the deployment of the images is the software applications happen.

With respect to their construction, which is reflected in Slide 32 of our presentation, again, what we're talking about is there's no inconsistency with our position. It's clearly what they're saying is virtual disk image files, that would satisfy the totality of that claim language, which says executable on a plurality of virtual machines. And that's what I'm trying to read out of the claim language.

Now, he talked about this two-phase capture process, which is reflected in the specification, and that's set forth in the -- starting on Slide 5 of our presentation. The reason that's important is because it gives, you know, the claim language indisputably requires two sets of images. That's on Slide 5, and it goes through this two-phase capture process. What that means is how do you create those specific image instances? And the reason that's important is it because it shows exactly how they're done.

And so you look on Slide 7, where it talks about virtual machine manager 402, that image is captured -- I'm looking out at the left-hand side, prior to the installation and configuration of a guest operating system and applications.

So what the asserted claims require is that the image instance be just other virtual machine manager. And so when petitioner's counsel says, well, we're, you know, we're misconstruing the term virtual machine as used in Khandekar. What they were doing and what those images show in Khandekar are reflected in what you see in Slide 11, which is the second part.

So phase one of the two-phase capture processes is just the virtual machine manager.

Phase 2 is the images. This is the software applications, and you see how they create and capture those images. If you run a virtual machine manager, looking at the left, the right-hand side of Slide 11 of our slides. You create a virtual machine, you install the operating system, you install the applications, you configure them, and then you capture that image.

So what they're talking about in Khandekar and Le is the totality of the image instance, right? The totality of the operating system and the applications. What they ignore more is there's no teaching. It's not in any place where you have an image instance of just a virtual machine manager stored separately so you can deploy that first, and then the image instance of the software applications. That's not disclosed anywhere in that except in the 138 patent.

And so when they try to pick and choose in their new argument that they deploy Le first, rebooted, and then deploy the image instances of the virtual machine, what they're doing is using hindsight to say, okay, this is what the 138 patent discloses. And now I'm going to try to find all those pieces in the prior art. And in our sur-reply, we actually provided an example of where Dr. Shenoy

had tried to describe that and said, well, you do this then subsequently you do this.

Well, that term subsequently is what's shown in our patent; not in anywhere in Khandekar and Le. And so if you look at Slide 13 of our presentation, you said these are where the term, this is a description in the specification. You do the step one, and then subsequently you deploy the image instances, the software applications onto the virtual machine.

So Dr. Shenoy the term subsequently, but showed nowhere in Khandekar, it's not shown anywhere in Le, and he inserts it into his description because but he's borrowing that language from the patent itself. And that's why we say in our sur-reply that's evidence of impermissible hindsight.

One other issue is that he -- petitioner's counsel said that the patent owner never addresses a inconsistent claim construction position in the District Court. There is no inconsistent claim construction position. There is, we, in this case, we didn't individually construe any of the claims. It's just the claim language as a whole with respect to the ordered sequence of steps.

With respect to the virtual machine manager, with it's very clear in the claim language

that there is a capability of that, 'cause the reason for that is because when it's first described in Claim 1, it's described as an image instance as part of the software image repository.

And I'm looking, you can look at any of the claim language, but with Slide 5 of our slides, it shows this language, and because you haven't executed anything yet, it certainly does have the capability that's 'cause that's where it's first defined. It's first defined in the software image repository.

And so it would be -- it wouldn't make any sense to say, well, you're executing it in the software image repository, because that's not where you deploy it, you deploy it onto the application nodes. And that's what's -- that's the second half, the Claim 1.

So there's no inconsistency whatsoever with respect to our positions in this case, and the patent owner's positions in the District Court litigation.

So, finally, go through the rest of these in order. The, you know, the patent, the petitioner talks about Khandekar and talks about it saying, well, it's simply using prior art methods to talk about its VMM, and that's true. And if you look at

Figure 1 of the Khandekar reference, it shows that prior art analysis, but what that prior analysis does not contain anywhere, and I think a good example of this is if you look on Slide 5 -- 40 of our presentation, it shows that Figure 1 of Khandekar, and it shows a VM and a VMM. What it's not showing anywhere is how you have two sets of images that you deploy one and then the other. And if you look at -- that's the prior art, right? And then what Khandekar does talk about is images of VMs, but it's only a single image. There's no discussion of two sets of images.

And petitioner's counsel talks about Slide 55 of our presentation where we show the advantages. This is not a misconstruction of anything. This is exactly why the method disclosing the 138 patent is so advantageous is because in Khandekar, you have to deploy the image as a whole.

Now, if you recall, petitioner's counsel said that -- he cited some parts of Khandekar that said you can customize it later. But in those examples that he was quoting from Khandekar, each one of them says that you install other applications later.

So that's exactly what this is. This does not -- our patent talks about image instances.

As I mentioned that the first part of my discussion, when you're installing something, that's a much different process than deploying by copying an image instance. When you're copying an instance, you have all the configuration already done. You have all the customized for how you want it so deploy it quickly. When you do an installation later, that is a much more time-consuming process and avoids the advantages of image instances.

So at the end of the day, there was no disclosure of Le or Khandekar where there was a deployment of two sets of images. It's not disclosed anywhere, and in while the patent owner's -- I'm sorry, the petitioner's expert will say, well, it was obvious to do this. Well, it's obvious if you read the 138 patent and look at it and say, oh, okay, I see the teaching. Now, I'm gonna go find that in the prior art. Where they say, well, you're just using VM software. Yes, the underlying software is the same, but how you use it, and how you create these two sets of images, and how you deploy them in series that is novel. And that's why we said in our sur-reply, just because the underlying VM software is commercially available, doesn't, you know, just because two painters paint the same picture using -- or paint the picture using

the same paint brush and paint doesn't mean you're gonna create the same picture. So what we've done is use those commercially available software, created a new way to store these images, and through that two-phase capture process, which is disclosing the first half of the first part of the slides in our deck, create those images, and deploy them in series where you deploy the images. This is the virtual machine manager. You create the emulated environment for the virtual machine, and then you deploy the image instances onto it. And that will create the flexibility that's reflected here in Slide 55.

Your Honor, I think I just have a few seconds left. So if you have any questions, I will be happy to answer them.

JUDGE BEGLEY: On your last point, are you saying that it's beyond the level of ordinary skill in the art to separately store the VMM? Is that what you're saying? That there's a gap in the petitioner's evidence and that a person of ordinary skill in the art couldn't store those images?

MR. YOUNG: I think a person of ordinary skill in the art would be capable of doing it. It wouldn't have a motivation to do it when you read the teachings of Khandekar and Le, because all of

them talk about a single step deployment.

They want to make it more efficient. In their mind, you deploy images once and that's faster. Then, in certain cases, you can customize them later.

So a person with ordinary skill would certainly be able to know how to do it. There will not be any motivation to do it. And that's true with many inventions. Many interventions, it's obvious to, or currently within their capability of doing it, but they wouldn't have any motivation to do it. And the only way they're able to piece this together with respect to the Khandekar and Le teaching is with the hindsight of the 138 patent, which is why I gave that example or that subsequently by petitioner's expert, because he was absolutely looking at the teachings of the 138 patents and then find those pieces in the prior art.

So to answer your question, yes, there's no question that a person with ordinary skill would be capable of doing it. They would not have a motivation to do it.

JUDGE BEGLEY: Thank you.

JUDGE DROESCH: I don't believe that the panel has any additional questions.

All right. So we can adjourn the hearing and go off the record at this point. Thank you.

(Proceedings conclude at 3:06 p.m.)

PETITIONER:

Harper Batts
Chris Ponder
Jeffrey Liang
SHEPPARD, MULLIN, RICHTER & HAMPTON LLP
hbatts@sheppardmullin.com
cponder@sheppardmullin.com
jliang@sheppardmullin.com

PATENT OWNER:

Daniel S. Young
Chad E. King
ADSERO IP LLC d/b/a SWANSON & BRATSCHUN LLC
dyoung@adseroip.com
chad@adseroip.com