Re: clubbing of windows vista and windows server2008 in a single boota

Source:
http://www.tech−archive.net/Archive/Vista/microsoft.public.windows.vista.general/2008−12/msg01766.html

- *From*: "Bill Yanaire" <bill@xxxxxxxxxxx>
- *Date*: Mon, 8 Dec 2008 11:42:05 −0800

Usually the girls go clubbing after 10 PM because there is more action after dark. I am not sure if you can go clubbing with Vista or Windows 2008 Server because they will think you are geeks and kick your ass out of the clubs.

Hope that helps.

"CDAC_BEST" <CDAC_BEST@xxxxxxxxxxxxxxxxxxxxxxxxx> wrote in message news:13E4BE1F−B976−40C7−9E5C−DA55155991C9@xxxxxxxxxxxxxxx

> i want to club the windows vista and windows server 2008 in a single bootable
> disk.
> please help for the above concern.
> i have already done the clubbing of images of .wim files.
> i failed in this process.
> As the internal files of both the OS is similar ...i need a another way out
> to extract the appropiate file from the single install.wim file .
>
> "CH" wrote:
>
>> Interestisng post and info. Apprciated the links you and *Mark VDBerg posted
>> a while ago.
>>
>> CH
>>
>> "MICHAEL" <u158627_emr@xxxxxxxx> wrote in message news:uVWcYS$rGHA.1596@xxxxxxxxxxxxxxxxxxxxxxx
>>
>>> http://www.apcstart.com/site/dwarne/2006/07/773/inside−vistas−new−image−based−install
>>>
>>> Vista's installation process is dramatically different to any previous

version of Windows: rather than being an 'installer', the install DVD is actually a preinstalled copy of Windows that simply gets decompressed onto your PC.

So how does it adjust to your hardware? How do you slipstream updates and drivers into it? Can you also 'preinstall' your favourite apps into your Vista DVD?

And most importantly, can you build a custom Vista install DVD that doesn't install all the 'free AOL trial' crap that typically comes bundled in with Windows?

We asked Microsoft Australia Technology Specialist for Windows Client, John Pritchard how it all works and got some surprising answers.

Dan Warne: Vista's "image based install" basically means that what you get on your Vista DVD is a preinstalled image of Vista, is that right?John Pritchard: Yes, what users' DVDs will contain is the install Windows Imaging (.WIM) file, which is basically our operating system folders wrapped up into one image file.

The users will put their DVD in, boot off it and run the setup and it will look to them like they are doing an install, but what it is really doing is grabbing the install.wim and executing that as an upgrade or clean install depending on what the user wants.

Dan Warne: So it's basically decompressing a preinstalled version of Vista onto the hard drive, and when you do an upgrade, it's basically putting a clean install of Vista on there and migrating your XP settings into Vista, right?

John Pritchard: Yes, that's right, it's a compressed image. We will ship it with fast compression, and then users just need to have the space on the hard disk for that image to be offloaded and decompressed.

There's also the advantage that it is file−based, not sector−based image, so you can install the image onto your hard drive without overwriting other data.

We also have advanced User State Migration with Vista. Users can take their settings from a previous version of Windows, migrate them off the PC and put them into an installable format for a new PC.

So, for example if they wanted to wipe their XP installation completely and start again with Vista, they could take their data off their XP installation with the User State Migration Toolkit and then restore it into Vista once they've completed their installation.

The User State Migration Toolkit can collect settings from Windows 2000 and XP SP.

Dan Warne: So is that something that ordinary consumers could use to migrate data from an old PC to a new Vista PC? Would it be easy enough for

consumers to use?

John Pritchard: Yes, it would be easy enough for consumers to use, though in that market there's also the Files and Settings Transfer Wizard.

James Bannan: I've used the XP Tool, the Transfer Wizard, a number of times for upgrading computers. The User State Migration Tool is more powerful but it is command−line based, so not as user friendly. You'd certainly find that power users would be drawn to it, definitely, especially as you can combine it with the WIM file image being a file based imaging format, meaning it's not an overwrite of your whole hard drive (unless you wish it to be).

Dan Warne: So in terms of the way the WIM system works, would it be possible to use WIM to back up, say, a Dell laptop completely as an image, and then restore it onto a Lenovo laptop with different hardware, for example? Would Windows be able to adjust to that different hardware?

John Pritchard: Yes, and that's one of the great benefits of it. The WIM format, being a file−based format, is separated from the hardware you're running it on. So you could take an IBM, Dell, Toshiba, whatever you've got, build your image up in it, and the way the traditional imaging process works, you can sysprep the machine, drop it and then create the image.

That way you can restore the image on multiple platforms. The caveat is that I wouldn't go from a 32−bit architecture to a 64−bit architecture, but staying inside 32−bit, you are no longer tied to the Hardware

Abstraction Layer (HAL) any more, and that is a great feature that releases us from so many challenges we've had in the past with HALs and multiple images.

You can now build your golden machine just like before, capture the image and then that image can be deployed widely and as you need to.

Dan Warne: what about keeping an image up to date. Users have had to get quite expert in doing this with XP because of its very out−of−date driverbase. Is this made easier with WIM?

John Pritchard: yes, you can update WIM images very easily.

There are two basic steps: one, you can just load a folder anywhere in the image you like. If there's something that requires a folder under the system32 directory that is completely unique to some particular hardware, you have the liberty to inject that folder into your WIM.

The other way is that you can use a DriverLoad utility, and that will actually place important things like disk drivers into their required location in the image, so when you are running a setup, it can look through its normal repository for drivers and bang, it's there, because it has been injected.

James Bannan: Out of interest, this all does rely on the image having been sysprepped, is that right? Because even though it is a similar deal with XP, even if the drivers are there, it does still need to run through that

setup process of assigning drivers to hardware. With WIM, I assume you couldn't just do a clean build, capture, inject the drivers, and drop it back on? It would still need to run through the driver allocation?

John Pritchard: With the actual released build of Vista, a user can mount the install.wim file on the Vista install DVD, mount it and put the drivers in themselves through the command line utilities.

When they unmount it, they'd have to burn another DVD of course, but they could have put drivers in there with it mounted into the file system. The drivers are actually injected into the right locations in there.

That's with an image that comes from Microsoft; if they want to build their own golden machine, they have to reboot it, boot into something like WinPE, and then use ImageX to capture the image, and once you've got that WIM image, you can inject drivers into it just like the Microsoft−supplied WIM.

Dan Warne: A lot of drivers nowadays come bundled up into EXE files that install everything into the right place for you. How would you inject those into a WIM image?

John Pritchard: You can actually do that with the unattend.xml file. You would put those EXE files on the disk and let the unattend process install them. If you look at the Windows System Image Manager, it has the capability to say, "look at these packages on a distribution

share, and
run these drivers as an application after you have built the system."

James Bannan: at what point in the install do those apps run?

John Pritchard: They're done in part seven, that's after the system has
been built, before logon. Now, with the EXE packaged drivers, you can
install them onto your golden machine, then build an image based on
that.
That's the other way of doing it, of course.

Dan Warne: I know that I have a cynical journalist's mind, but isn't
that
a bit of a risk for malware to be injected into Vista install DVDs,
given
that those apps are executed before logon?

John Pritchard: Yes, well I would certainly recommend when people are
looking at any content they make sure they have the approved and
hologrammed DVDs to make sure they're dealing with the genuine product,
to
get away from not knowing where the source comes from. But if they have
got control of the unattend and built it themselves then hopefully they
know what they are putting on it.

James Bannan: plus I believe ImageX itself can do a verify on a WIM so
I
guess that is an advantage if you have got the original WIM, a
corrupted
WIM won't match up to the original.

Dan Warne: I guess like any software that can be corrupted, people will
just have to go back to the original hashes.

John Pritchard: I think it comes back to people having the original
software first, and that is the level of assurance I would look

for.

Dan Warne: I guess I was thinking more of a corporation that might have
a
WIM image sitting somewhere on a network share and a rogue employee
might
go in and add something to the image.

James Bannan: it's probably a bit too much to rely on WIM to be able to
protect itself from rogue IT administrators. you're asking a lot.

Dan Warne: yeah, I guess if you have file access you can do pretty much
whatever you like can't you.

James Bannan: pretty much.

John Pritchard: Also with larger enterprises they'll have something
like
SMS, and the users don't see that. It's deployed under SMS like an
application. it's managed centrally and that has very good process
around
that to protect corporate WIM images.

James Bannan: So could you inject the Office installation files into
your
WIM, and could you have different installs for different machines,
based
on different unattend.xml files for example?

John Pritchard: Certainly, and this is where you're getting into
leveraging not only the unattend file, but also the Windows System
Image
Manager. You can set up all your applications as packages, so you can
have
one unattend file that installs office, and another that doesn't. An
unattend file can do patches, drivers and applications, effectively

simulating a GUI run−once.

James Bannan: I guess then, if your home user who is interested in this kind of thing but doesn't have access to WDM or SMS, they're just going to have to customise a number of unattends and specify the one they want when they do the build.

John Pritchard: Yes, and if you want to build your own DVD and put your unattend file into the root of the DVD, there's only one option there. It's called autounattend.xml − it has to be that name because it's what the build process looks for. So if you wanted to have various unattended installations, you'd just have to manually switch those files yourself.

James Bannan: I guess though you could probably have an open−source PXE if you wanted to.

John Pritchard: That one I don't know about.

Dan Warne: [sarcastic] open source is the enemy, James!

James Bannan: [laughs] yes but Microsoft is interested in how its software integrates with everything else, surely.

John Pritchard: It's always good there to hear from what our customers are saying and what they need.

Dan Warne: What about the process of updating the Vista image with service packs and patches? The process for slipstreaming in XP is relatively straightforward once you know how but it isn't exactly intuitive, or as easy as running Windows Update.

John Pritchard: Well, in Vista, we can do that once the machine is built and on the network; you can use WSUS, or if they have an SMS environment you can patch the deployed machine in either of those two ways, so that doesn't change.

But once you build an image, it poses a problem because it's likely to be out of date as soon as you close it off. So, with that, you can take the image, and say, "OK, I'll build a command line file that enables me to mount the image, apply the images to the OS while it is mounted, and then seal up and commit the changes to the image, and distribute the image."

Dan Warne: So is there an automated way to grab all the patches off Windows Update and automatically apply them to an image? Or would you have to download each patch individually and manually apply them?

John Pritchard: You've got the image effectively mounted as a file system, so you'd apply the patches as command−line patches. You would have to get each patch and apply it. It's like slipstreaming SP2 into an SP1 installation.

But if you have an image that's, say 2.5GB, instead of patching it and having to push that entire image file out to different file shares, what you can do is instead of sending out the whole patched image again, you simply make your patch commands and then just send out

the command line
to
mount the image and apply the patches locally and unmount
the image. So
at
each point, they can run a series of batch files to update their
image.

Dan Warne: So, in terms of customising the Vista install
DVD to remove
software components. Because inevitably in a large operating
system
there's
a lot of stuff in there that people don't want or use, like in
XP, the
MSN
Browser. Is there an interface for configuring WIM that is a
bit more
componentised, rather than just looking at the files on the
disk? Can
you
actually select apps in Windows and just get them ripped out
of the
image?

John Pritchard: Yes, where I'd go to for that is if you take the
Microsoft
DVD that will be shipped out, we again go back to the
unattend.xml and
you
can build an unattend.xml that says, "I want this, I want this,
I want
my
partitions configured like this, do all that but also select that
you
want
this game, but not solitaire, or whatever."

You can then put that unattend.xml file on a USB key and if
you plug
that
in when Vista is installing, it will base its install process on
the
unattend.xml instruction file. It means that you don't have to
build a
custom DVD for a custom install. Consumers can take the
System Image
Manager, build up the unattend as they would like, put it on a
USB key
and
use that to install from the Microsoft−standard image file.

Dan Warne: Cool, so that's presumably a new feature in Vista? I knew you could script Windows installations previously, but you've never been able to run that script from a USB key, right?

John Pritchard: Yes, that's right. This is where we've got the ability to look for the USB port. It's like having a WINNT.SIF file being looked for in the root of a floppy drive. What I do for my customers is they have the bootable Vista build DVD and they put their unattend on a USB key, which saves them having to rebuild their DVDs all the time.

James Bannan: a lot of corporate customers more than likely have the facilities to be able to install off a network share, won't they. It's a fairly safe guess that most power users would have more than one computer at home. You'd have your file server, or something along those lines, so you wouldn't have to go to the length of having a USB key, would you. You could just have the unattend.xml in the share root and launch the installation from there, is that right?

John Pritchard: Yes, you can do, if you boot up under something like WinPE, because you obviously have to be able to get to the share, get an

.