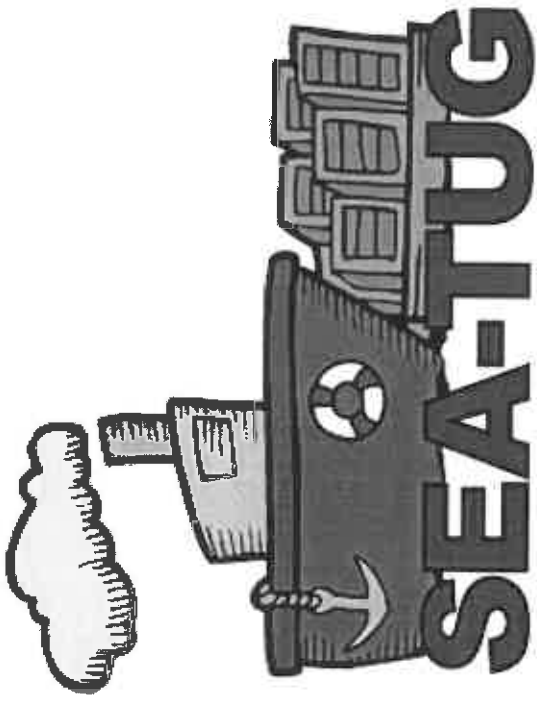
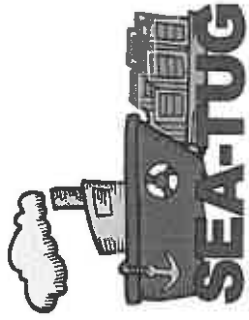


WELCOME TO

Systems
Engineering &
Administration
Technology
User
Group



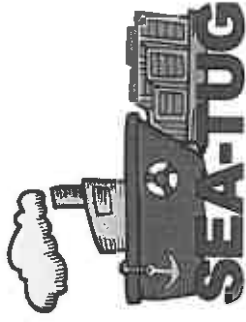
Wednesday, April 19, 2017



bringing **IT** togetherSM

Agenda

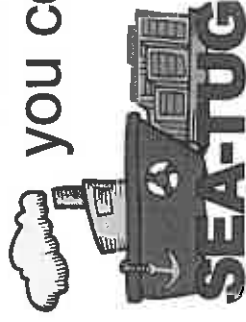
- Introduction
- Main Presentation
- Q&A



bringing **IT** togetherSM

Introduction to SEA-TUG

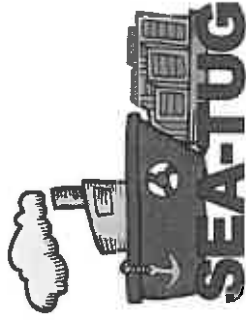
- Founded in 2001 by Rob Bergin and Steve Noel to help IT professionals in the seacoast collaborate and enhance their knowledge
- New steering committee formed July 2016
 - John Whelan, Pamela Capper, Deb Gale, PJ Soucy, Joel Wright, Derek Rolfe, Rob Maciorski, Terry Jamro, Chris Morris
- Part of larger user group communities
 - Boston User Groups, Meetup, etc.
- This is YOUR user group – help us make it better.
What topics do you want covered? What resources can you contribute?



bringing **IT** togetherSM

Thank you to...

- Alexander Technology Group
 - Food/beverages
- Great Bay Community College
 - Meeting space

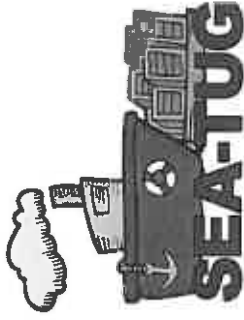


bringing **IT** togethersm

Tonight's Presentation

Building a Windows Client Image With MDT

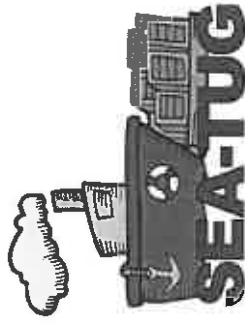
Tobin Weltin



bringing **I**T togethersm

Microsoft Deployment Toolkit

- MDT can help build an automated installation source for deploying client OS and Office from a single machine or a central server distribution tool, such as WDS or SCCM. Device drivers, Windows updates and software can be included with the build.
- <https://technet.microsoft.com/en-us/windows/dn475741>



bringing **T**ogetherSM

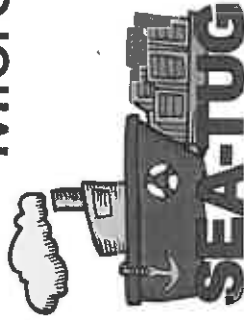
- All the software intended for installation (Operating System, drivers, updates and applications) are added to a pool of available software and packaged into *deployment packages*.
- MDTToolkit generates a custom Windows PE image that allows client machines to install the assembled deployment packages over the network or from a CD/USB stick



bringing **IT** togetherSM

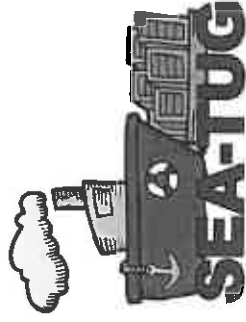
- **There are three types of deployments:**
 - **Zero Touch Installation (ZTI)**
 - ZTI is a fully automated deployment which requires no user interaction whatsoever. Requires a persistent connection to the distribution point.
 - **Lite Touch Installation (LTI)**
 - LTI deployments require limited user interaction, and needs very little infrastructure, so it can be installed from a network share, or portable media.
 - **User Driven Installation (UDI).**
 - UDI deployments require full manual intervention to respond to every installation prompt.

- **ZTI and UDI deployments both require a Microsoft System Center infrastructure.**



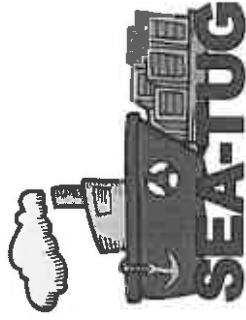
Windows Deployment Services

- The successor to Remote Installation Services, WDS is intended to be used for remotely deploying Windows clients but also supports other operating systems (unlike RIS).
- WDS uses disk imaging, in particular the Windows Imaging Format (WIM).
- WDS is included as a *Server Role* in all 32-bit and 64-bit versions of Server 2008 and up.



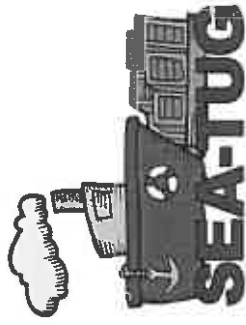
bringing **IT** togethersm

- WDS works in conjunction with the Preboot Execution Environment (PXE) to load a WinPE environment for install and maintenance tasks.
- WDS functions as both a storage repository for the PXE network boot images and a repository for the actual operating system images to be installed on the target computer.
- When multiple boot images are available, PXE booting via WDS will present the end-user with a boot menu to select the image to load.

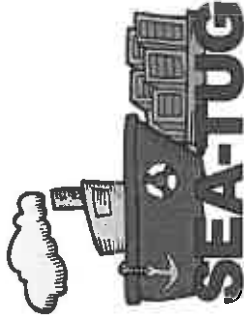


bringing **IT** togethersm

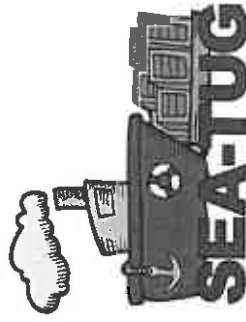
- Two special scripted Windows PE boot images can be created which automate these tasks.
- These images are created using WAIK, in combination with Windows 7/10 installation media containing the source WIM images, then added to the WDS server's boot image repository.
- A difficulty of Windows PE booting is that it needs to include network drivers and disk controller drivers intended to work with the target hardware to be imaged.



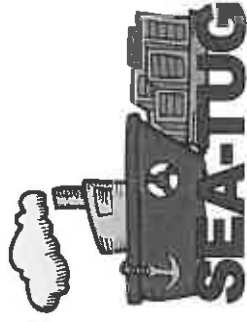
- The process of adding drivers to the Windows PE boot image can be automated using the WDS server console:
 - Select the source WIM image, which may be either a new one created from original Windows 7 installation DVDs or a previously configured WIM.
 - Select the drivers to install into the WIM
 - WDS mounts the WIM to a virtual path, adds drivers to the virtual path, and generates a new WIM
 - The updated WIM image is added to the boot image section of the WDS repository
- This process can be repeated at a later time when a new system type needs to be captured



- Applying a captured image involves running a second Windows PE "Apply" boot image on the target system to receive the image.
 - This boot image also needs the appropriate network and disk controller drivers as with the Windows PE Capture boot image.
- The system is booted using PXE network booting and the Windows PE Apply image is loaded.
 - Operator logs on and selects a boot image to apply.
 - A disk partitioning screen appears and the location for the target image is selected.



- The image data is applied and the system reboots, either running the Sysprep manual mini-setup process or following the script created during the initial Sysprepping.



bringing **T**ogethersm

Microsoft Deployment Toolkit

TOBIN WELTIN

TOBIN.WELTIN@GMAIL.COM

The deployment landscape

- ▶ There are a ton of imaging and deployment solutions out there!
 - ▶ HDD cloning
 - ▶ Fat-Image burning
 - ▶ Networked OS Deployment solutions (MS WDS, MDT, Ghost, Glazier)
 - ▶ Full Endpoint Management Suites (LANDesk, SCCM, KACE)
 - ▶ Software-only deployment (PDQDeploy, Powershell, etc)

Which method(s) do you use in your organization?

What features did you prioritize?

Some Imaging decision points

- ▶ Fat vs Thin images
- ▶ Hardware Independence (HII)
 - ▶ Driver matching via PnP or WMI?
- ▶ File-based vs. Sector-based images
- ▶ Transmission capability and protocol
- ▶ Windows or OS agnostic?
- ▶ PE Environment delivery

Enter MDT

What MDT can do for you

- ▶ Get you down to that one Golden Image
- ▶ Configure Desktops, Laptops and Tablets in one single workflow
- ▶ Move user profiles to new systems, including application settings
- ▶ Rudimentary self-service Application Deployment
- ▶ Install Windows Server roles
- ▶ Use PXE or Discovery Media

What MDT cannot do for you

- ▶ Anything Open-Source
- ▶ Push deployments.
- ▶ Reporting is very basic
- ▶ Multicast Deployment (out of the box)
- ▶ Activate Windows with OEM Licenses

Installing MDT

- ▶ Latest version is Build 8443. They stopped mirroring SCCM's version numbers.
- ▶ Requirements:
 - ▶ Most recent Assessment and Deployment Kit (Windows 10 1703 ADK)
 - ▶ Deployment Tools
 - ▶ Windows Preinstallation Environment (Windows PE)
 - ▶ User State Migration Tool (UMST)
 - ▶ VLSC Windows client OS install media.
 - ▶ Windows Deployment Services is recommended for PXE and Multicast capability, but not required.

Building your Golden Image

- ▶ Golden Images are your endpoints stripped down to their most basic configuration.
- ▶ Don't set anything that can't be universally adopted by your organization. This includes updates.
- ▶ Do:
 - ▶ Install updates from WSUS and activate features
 - ▶ Install Microsoft Applications (IE, Windows Live 2012)
 - ▶ Build Image on Hyper-V (Gen 1 for 7, Gen 2 for >7)
- ▶ Be careful:
 - ▶ Installing 3rd party applications, especially Antivirus or Agents
 - ▶ Running any programs while building image.
 - ▶ Building Image on actual hardware or VMWare, Xen, QEMU.

Sysprep and DISM

- ▶ MDT can be used without knowing either of these.
- ▶ Sysprep is run on a running OS to prepare it for deployment. There are three modes:
 - ▶ Audit Mode
 - ▶ OOBE Mode without Generalize
 - ▶ OOBE Mode with Generalize
- ▶ DISM is run against offline images stored as WIM files. It can:
 - ▶ Install Updates and Drivers without the OS being turned on.
 - ▶ Apply Unattend.xml files to offline images.
 - ▶ Mount Offline Filesystems for modification, configuration.

Capturing your first image in MDT

- ▶ To create a Capture task, you need to already have an OS to import. Kind of a “chicken or the egg” question, right?
- ▶ You can get around this by importing the install.wim file from Windows OS Install media. You may need to convert .esd files to .wim with DISM.
- ▶ You also need to create Boot media and put it on a PXE Server, or burn it to a USB stick.
- ▶ The Capture task creates a WIM file on your Deployment share. You can import this into your server for future deployment.

Getting ready to deploy: Drivers

- ▶ There are a few ways to get drivers on to images during deployment.
 - ▶ Pre-deployment injection into mounted WIM with DISM
 - ▶ In-Band injection with PNP code matching
 - ▶ In-Band injection with WMI and Driver Group matching
- ▶ Potential Pitfalls
 - ▶ The wrong driver can result in some really funky behavior
 - ▶ Slightly different drivers have the SAME PNP code
 - ▶ Drivers can take up a lot of room
 - ▶ The wrong driver can stop your deployment in its tracks.

Wmic computersystem get mod

Getting ready to deploy: customsettings.ini

- ▶ The “Secret Sauce” of MDT. This is where the extensibility of the product becomes apparent.
- ▶ Used to set deployment variables based on any criteria gathered by MDT including Task Sequence, computer type, make and model, serial number, default gateway, etc.
- ▶ Also used to skip steps in Lite Touch Deployment by defining variable settings and setting a “skip” setting.

Getting ready to deploy: Task Sequences

- ▶ The defaults are pretty good for everything you'd want to do, with some few exceptions.
- ▶ For WMI Matching, you need to set the DriverGroup001 variable to however you want to match your hardware to a driver group.
- ▶ If you want to read the customsettings.ini file into the Task Sequence, you have to specify it on the "Gather Local" tasks.
- ▶ Instead of specifying all your rules in the customsettings.ini, sometimes it's easier to just delete or disable extraneous tasks in the Task Sequence.

Resources

- ▶ Deployment Research > Research
 - ▶ <http://deploymentresearch.com/Research/Post/325/MDT-2013-Life-Touch-Driver-Management>
- ▶ Deployment Research > Research
 - ▶ <http://deploymentresearch.com/Research/Post/322/Back-to-basics-Understanding-Unattend-xml-automation-in-Windows-7>
- ▶ The Complete Guide to Preparing a Windows 8 Deployment Image using Audit Mode and Sysprep with an unattend.xml Answer File
 - ▶ <http://sybaspot.com/the-complete-guide-to-preparing-a-windows-8-deployment-image-using-audit-mode-and-sysprep-with-an-unattend-xml-answer-file/>

Resources

- ▶ [Migrating from Windows XP to Windows 8.1 using MDT 2013](#) – Michael Niehaus' Windows and Office deployment ramblings
 - ▶ <https://blogs.technet.microsoft.com/mniehaus/2014/01/09/migrating-from-windows-xp-to-windows-8-1-using-mdt-2013/>
- ▶ [MDT Log file locations](#)
 - ▶ <https://social.technet.microsoft.com/Forums/en-US/b7cbd22f-3a79-43a0-97b7-88503396b89a/mdt-log-file-locations?forum=mdt>
- ▶ [Deploying Windows 7 - Part 26: Managing Drivers – By Make and Model - TechGenix](#)
 - ▶ <http://techgenix.com/Deploying-Windows-7-Part26/>

Resources

- ▶ [MDT] MDT Resources
 - ▶ <http://www.edugeek.net/forums/o-s-deployment/76099-mdt-resources.html#post678279>
- ▶ Customising bootstrap.ini in MDT – IT Support Guides
 - ▶ <https://www.itsupportguides.com/knowledge-base/configmgr-sccm/customising-bootstrapini-in-mdt/>
- ▶ Dell download packs
 - ▶ <http://en.community.dell.com/techcenter/enterprise-client/w/wiki/2065.dell-command-deploy-driver-packs-for-enterprise-client-os-deployment>

Resources

- ▶ This is a sample of a customsettings.ini and shows examples of the logic you can put in.
- ▶ <https://mdtguy.files.wordpress.com/2013/10/cs.png>

```
1 [Settings]
2 Priority=ByLaptopType, ByDesktopType, Default
3 Properties=MyCustomProperty
4
5 [ByDesktopType]
6 Subsection=Desktop-!IsDesktop%
7
8 [ByLaptopType]
9 Subsection=Laptop-!IsLaptop%
10
11 [Desktop-True]
12 OSDComputername=W7PC-!Right-!Right (!%SerialNumber%,4) #
13 MachineObject00=00=Desktops, 00=Workstations, DC=Spudco, DC=com
14
15 [Laptop-True]
16 OSDComputername=W7LT-!Right (!%SerialNumber%,4) #
17 MachineObject00=00=Laptops, 00=Workstations, DC=Spudco, DC=com
18 Applications001={237547b0-10bb-4053-8737-2f40babffe98}
19 Applications002={81e55836-cefa-4b53-b517-70df7913a1b8}
20
21 [Default]
22 _SMSOrgName=Running $TaskSequenceID% on %OSDComputername%
23 SkipBDWWelcome=YES
24 SkipApplications=NO
25 Applications001={64537db0-e234b-4e73-e791-2ca8594b38e2}
```

```
26 SkipComputerName=YES
27 SkipTaskSequence=NO
28 OSInstall=Y
29 SkipAppsOnUpgrade=NO
30 SkipAdminPassword=YES
31 SkipDomainMembership=YES
32 SkipUserData=YES
33 SkipComputerBackup=YES
34 USMTMigFiles001=MigApp.xml
35 USMTFiles002=MigUser.xml
36 ScanStateArgs=/v:s /o /c /ue:* \* /uel:90
37 LoadStateArgs=/v:s /c /lac
38 SkipTimeZone=YES
39 SkipLocaleSelection=YES
40 TimeZoneName=Mountain Standard Time
41 SkipBitLocker=YES
42 SkipCapture=YES
43 SkipRoles=YES
44 SkipSummary=YES
45 SkipFinalSummary=NO
46 SkipPackageDisplay=NO
47 DriverSelectionProfile=Nothing
48 DriverInjectionMode=ALL
49 DriverGroup001=Windows 7\x64\!Make%!\Model%
50 HideShell=YES
```

Questions?