

Creating a Dynamic Client Build Using Driver Management

Before developing our current ITCB process, it took us six months to release a new build because we had to manually create builds for each platform we support. Now, with our streamlined ITCB process, we can release a build—including changes to the application stack—every 10 weeks.

John Gonzalez
IT Client OS Product Line Manager, Intel IT

Justin Phillips
Systems Programmer, Intel IT

Johnnie Rodriques
Systems Programmer, Intel IT

Executive Overview

Intel IT's modular IT Client Build (ITCB) process enables us to efficiently manage the thousands of laptop and desktop PCs—and more than 2,000 software applications—used by Intel's 100,000-plus employees. At the time of publication, Intel IT officially supports 61 make-and-model combinations of PCs from 10 unique hardware suppliers—resulting in about 75 combinations in total. We require the installation of 30 applications on each PC at Intel, such as virus scan, VPN, whole disk encryption, and office productivity software. Before developing our current ITCB process, it took us six months to release a new build because we had to manually create builds for each platform we support. Now, with our streamlined ITCB process, we can release a build—including changes to the application stack—every 10 weeks.

In our dynamic global enterprise environment, we need to efficiently integrate our core applications with a wide variety of hardware platforms. By using a standard configuration of mobile business PCs featuring the Intel® Core™ processor with Intel® vPro™ technology, Intel® Solid State Drives (Intel® SSDs), and Windows 7*, we have created a stable computing environment that supports our streamlined ITCB process.

The primary components of the ITCB process are as follows:

- Boot media that is used to bootstrap the build process
- An image that contains the base OS and core applications that apply to the majority of the user population

- An extensive device driver library
- A set of platform packs, each of which contains drivers, utilities, and firmware specific for each make and model of PC running on a supported platform

Our streamlined ITCB process increases IT efficiency and velocity—our ability to deliver solutions faster. It helps us manage our global PC fleet so we can update all supported platforms at once. We can also, when required, use the same process to successfully build platforms that Intel IT does not officially support, such as engineering prototypes and evaluation units.

Contents

Executive Overview	1
Business Challenge	2
Solution.....	2
IT Client Build (ITCB) Process Overview.....	2
Boot Media.....	3
Driver Management.....	3
Platform Packs.....	4
An Inclusive Build Process.....	4
Results.....	5
Conclusion.....	6
Acronyms.....	6

IT@INTEL

The IT@Intel program connects IT professionals around the world with their peers inside our organization – sharing lessons learned, methods and strategies. Our goal is simple: Share Intel IT best practices that create business value and make IT a competitive advantage. Visit us today at www.intel.com/IT or contact your local Intel representative if you'd like to learn more.

BUSINESS CHALLENGE

With more than 100,000 employees using more than 2,000 applications on one or more PCs to perform their jobs, Intel's PC fleet is large and complex. At the time of publication, Intel IT officially supports 61 make-and-model combinations of PCs, 10 unique hardware suppliers, and two OSs—resulting in about 75 combinations in total. The core application stack consists of 30 applications that must be installed on each computer at Intel, such as virus scan, VPN, whole disk encryption, and office productivity software. In our dynamic global enterprise environment, we need to efficiently create a cohesive unit out of the hardware, OS, and enterprise applications.

In addition to the officially supported PC makes and models, there are numerous non-standard systems in use across Intel, such as engineering prototypes and evaluation units. Although not officially supported by Intel IT, these non-standard systems have the same basic requirements as the systems we support officially—they all need a working OS and they must be compliant with information security policies because they are on the Intel network instead of in isolated labs.

In the past, we created a different build for each hardware platform. When we needed to update a version of software, we had to repetitively update each of those builds with exactly the same change. Clearly, we needed an easy, quick, and cost-efficient way to build and upgrade both standard and non-standard platforms.

SOLUTION

By using a standard configuration of mobile business PCs featuring the Intel® Core™ processor with Intel® vPro™ technology, Intel® Solid-State Drives (Intel® SSDs), and Windows 7*, we have created a stable computing environment that gives Intel employees a positive computing experience. A more significant benefit: This computing environment also supports our streamlined IT Client Build (ITCB) process—increasing IT efficiency and velocity in managing our global PC fleet.

We have developed a single build process where, if a configuration changes, we can update all platforms at once. This makes IT more efficient at deploying builds to all our supported hardware platforms.

IT Client Build (ITCB) Process Overview

Deciding what image and boot media to use and how to manage drivers and platform packs were part of developing our build process. We had three build images to choose from.

- A thick image typically contains the base OS, plus every application that might be used in every usage scenario of that build. A driver library representing every supported platform is often also included. In physical disk size, this build type creates a very large image because everything is included; a change to any application affects every platform and image.
- A thin image is typically made up of only the base OS and boot-critical drivers. This type of image is very flexible but requires every application, even if it is used by every employee, to be installed in a separate process, which can be repetitive and inefficient.
- A hybrid image takes the best features of thick and thin images.

The ITCB process uses a hybrid image, which includes the base OS, Windows in-box drivers¹ required for a particular platform, and core applications that apply to 80 percent or more of the user population.

We augment the image with other drivers from an extensive device driver library, along with a platform pack for each officially supported platform. Each platform pack includes a subset of drivers, utilities, and firmware specific to that platform. Using the ITCB process, illustrated in Figure 1, we can build a platform in as little as 60 minutes.

Boot Media

To bootstrap the build process, we use boot media based on Windows Preinstallation Environment* (Windows PE*) 3.0, which is the lightweight version of Windows 7*. Boot media can be a CD, USB flash drive, or a network copy accessed through the Windows Preboot

¹ Microsoft uses the term in-box driver to refer to any driver that is built-in to Windows 7*.

eXecution Environment* (Windows PXE*). Regardless of the way the boot media is loaded, identical content is used in every case; only the delivery method changes.

Driver Management

To make our ITCB process as flexible as possible, we take a unique approach to driver management. We focus first on boot-critical device drivers that are required for a successful build. Then we use a process called driver injection, which copies the required information files (INFs) necessary to support devices found on the system at build time, and one or more reference drivers.

For most systems to successfully complete the build process, the target system must have a mass storage controller (MSC) and a network interface card (NIC) that Windows 7, as released, supports or that one of the many driver packages we integrate into the ITCB process supports.

BOOT-CRITICAL DEVICE DRIVERS

Generally, only two categories of boot-critical devices are required to build a Windows-based system using the ITCB process:

- **MSC.** Enables writing to the hard drive, so we can install the OS on it
- **NIC.** Enables connecting to the network, so we can access the files and services necessary to complete the build process

When we build a platform, the build process copies all the MSC and NIC drivers included with Windows PE for all the devices supported by Windows 7 at the time Microsoft released it. To increase the flexibility of our build process, we also include a driver set for an external NIC that can be connected to almost any system using the USB port. By adding this driver set, we can successfully build, manage, and secure almost any PC.

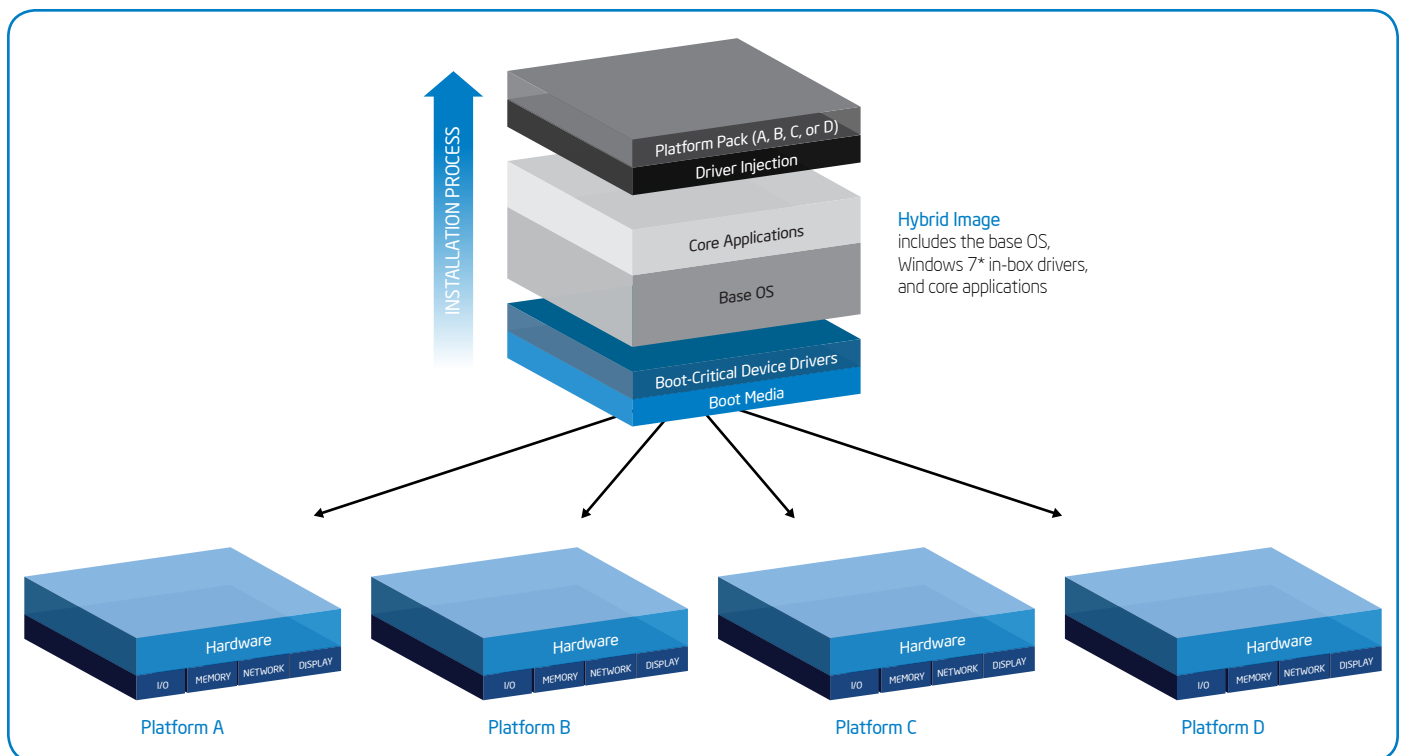


Figure 1. The IT Client Build process, which includes a hybrid image combined with drivers from our extensive device driver library and platform packs, supports a wide variety of makes and models of PCs.

DRIVER INJECTION AND REFERENCE DRIVERS

We maintain a large driver library with drivers obtained from two sources:

- Device manufacturers, such as the company that manufactures the MSC or the graphics adapter
- OEMs, the companies that sell the PCs

Just before OS setup, we scan the target platform for devices and use driver injection to copy any matching drivers from our driver library. Some of these drivers supersede existing drivers in Windows PE, while others are completely new.

For certain categories of devices, we use reference drivers, which are drivers sourced from the device manufacturers, not from the OEMs. We have found that device manufacturers update their drivers more often than OEMs do. To prevent the majority of platform performance and stability problems we observed with older drivers, we always get the most up-to-date versions of the following drivers:

- MSCs
- NICs
- Graphics adapters
- Base (motherboard) chipsets

Unlike OEM drivers, which often work with only one OEM-supplied device, reference drivers support many different generations of individual devices. For this reason, we can use reference drivers to build some devices that are not found in Intel IT standard platforms.

The broad compatibility of reference drivers means that we can use a single reference driver to support many different makes and models of PCs, instead of having a separate driver for each combination of make and model that contains the device supported

by the driver. This significantly reduces the number of driver versions that we have to manage, support, test, and deploy.

We have experienced a few minor feature differences when using a reference driver instead of an OEM driver. For example, on some models of PC from a particular supplier, we found that the reference driver, in combination with some models of internal Intel® graphics chipsets, prevents the on-screen brightness indicator from appearing; it does appear when using the OEM driver. These types of issues have been sufficiently minor and infrequent that the overall value of using reference drivers has far outweighed its flaws.

Platform Packs

Simply installing the drivers identified as missing in the Windows Device Manager is not enough to create an optimized system. Windows 7 includes in-box drivers for a wide range of devices. Some of these are generic drivers, such as Generic PnP Monitor, and some are device-specific, such as Intel® PRO/1000 T Desktop Adapter.

By design, most in-box drivers give only the base functionality to accommodate the broadest range of devices and configurations. To get more features and better performance, it is common practice to get a driver package from the device manufacturer. Also, because of the nature of the OS release cycle, by the time Windows ships, many of the drivers included in-box for newer devices are already outdated and can cause problems with stability, performance, and battery life. This is why we create a platform pack for each platform we officially support, which we install after we have installed the Windows and reference drivers, so it is not part of the actual build image.

Typically, a platform pack includes every driver provided by the system's OEM, utilities, and firmware, such as BIOS and Intel® Active Management Technology firmware. In contrast, we rarely install all of the utility software that comes with the system; instead, we choose to install only those utilities we believe gives our customers an optimal computing experience, reflecting an ideal balance between PC stability, performance, and battery life.

An Inclusive Build Process

We have designed the ITCB process so that it does not preclude the building of any computer. The successful build of a system is purely a function of whether we support the appropriate boot-critical devices. Using the ITCB process, we have successfully built netbooks, tablets, convertibles, slates, engineering prototypes, and even specialty devices such as wearable computers and digital signage devices.

While platform packs typically don't exist for non-standard platforms, in certain circumstances installing an existing platform pack can minimize the amount of manual driver installation. If the build process is unable to find a platform pack that matches exactly the make and model of the PC being built, Intel IT can choose a similar platform pack from a list of all platform packs.

In general, we recommend choosing a platform pack that is for a PC from the same OEM and in the same platform family. This increases the likelihood that the platform pack contains similar, or even identical, drivers to those needed by the non-standard platform. We do caution users, however, that for non-standard platforms, the selected platform pack is not considered tested or certified and is not officially supported by

Intel IT. Regardless, the non-standard system can connect to the Intel network, and we support standard office productivity software issues, so long as the hardware isn't the cause of the problems.

In the case of non-standard platforms, after the build is complete, we may need to optimize the system manually by using the Microsoft Windows Update feature to install any updated drivers that are available, and by visiting the OEM's web site to locate missing or updated drivers for the specific make or model of PC being built.

RESULTS

The ITCB process works so well that we can complete a build release cycle in less than half the time it used to take.

The ITCB process, including the use of mobile business PCs with a standard configuration offers several benefits.

- Improves IT's velocity—our ability to deliver solutions faster. This improves time to market and Intel's overall competitiveness.
- Increases IT efficiency by being able to do more in less time with fewer resources and fewer errors and incidents.

- Supports a global environment through automation. The build can be built in any geography, and all results are identical and consistent.
- Supports IT consumerization and a wide variety of form factors.

As shown in Figure 2, before developing our current ITCB process, it took us six months to release a new build because we had to manually create builds for each platform we support. With our modular and dynamic ITCB process, we can release a build—including changes to the application stack—every 10 weeks. Figure 3 shows our improved release cycle, which includes the following:

- Three weeks of engineering
- One week of validation
- Five weeks of quality assurance (QA)
- One week to release to production

We have also created a streamlined five-week version of this process for builds that include only new hardware, such as our deployment of Ultrabook™ devices. In this compressed version, the schedule calls for two weeks of engineering, one week of validation, and two weeks of QA.

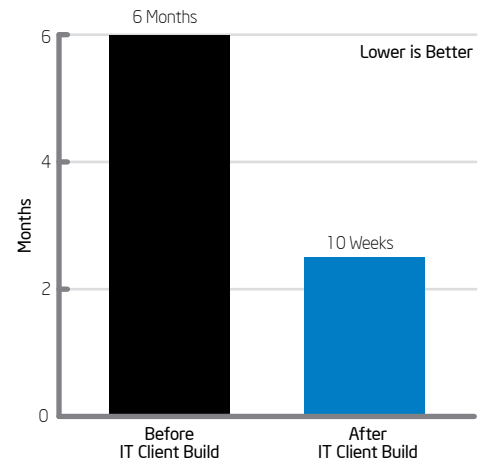


Figure 2. The IT Client Build process reduced the time it takes to release a build from six months to 10 weeks.

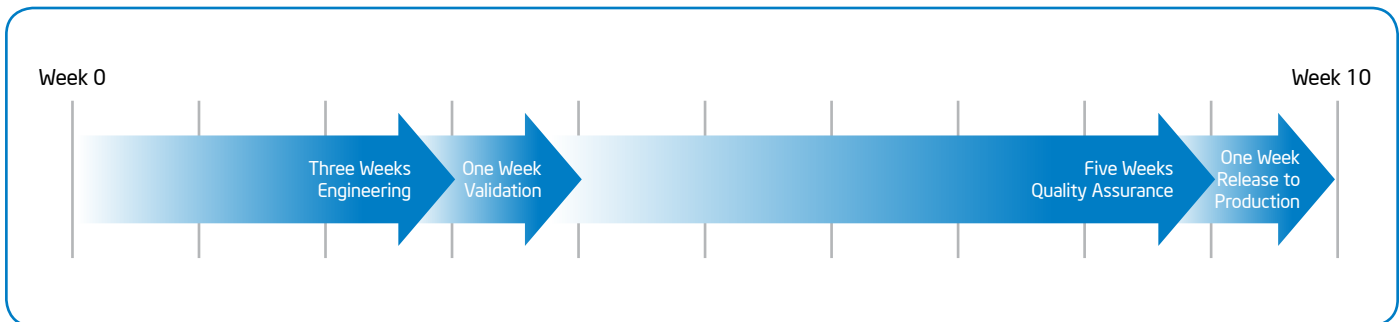


Figure 3. Our release cycle includes three weeks of engineering, one week of validation, five weeks of quality assurance, and one week of release to production.

CONCLUSION

By using a standard configuration of mobile business PCs featuring the Intel Core processor with Intel vPro technology, Intel SSDs, and Windows 7, we have created a strong computing environment that supports dynamic, modular management of Intel's large and complex PC fleet.

We bootstrap our ITCB process with an up-to-date copy of the boot media and install an image that contains the base OS and core applications that apply to the majority of the user population. Using driver injection, we load the following drivers from our extensive device driver library:

- MSC and NIC boot-critical drivers, as well as a universal USB NIC driver that works with practically any platform with a USB port

- INF-based drivers for devices found on the system
- A set of up-to-date reference drivers obtained from the device manufacturer for MSCs, NICs, graphic adapters, and motherboard chipsets

On top of these, we install the appropriate platform pack that contains make- and model-specific drivers, utilities, and firmware for a particular system. With this modular approach to client build, we can build almost any PC in a secure and managed way using a single image and build process.

ACRONYMS

INF	information files
ITCB	IT Client Build
MSC	mass storage controller
NIC	network interface card
QA	quality assurance
SSD	solid-state drive
Windows PE*	Windows Preinstallation Environment*
Windows PXE*	Windows Preboot eXecution Environment*

For more information on Intel IT best practices, visit www.intel.com/it.


This paper is for informational purposes only. THIS DOCUMENT IS PROVIDED "AS IS" WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NON-INFRINGEMENT, FITNESS FOR ANY PARTICULAR PURPOSE, OR ANY WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, SPECIFICATION OR SAMPLE. Intel disclaims all liability, including liability for infringement of any patent, copyright, or other intellectual property rights, relating to use of information in this specification. No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted herein.

Intel, the Intel logo, Intel Core, Intel vPro, and Ultrabook are trademarks of Intel Corporation in the U.S. and other countries.

* Other names and brands may be claimed as the property of others.

Copyright © 2012 Intel Corporation. All rights reserved.

Printed in USA
0312/JGLU/KC/PDF

 Please Recycle
326412-001US

