

Managing the Software Packaging Process— A Crucial Part of the Application Lifecycle

Many IT departments that invest heavily in large-scale tools to deploy business applications often overlook the need to properly prepare applications before deployment. Understanding the pre-deployment workflow is crucial to tracking and reporting on the status of the process. This white paper introduces and defines the roles and responsibilities of three classes of users who touch the pre-deployment process:

The Requestor – This user defines the application needs and submits a request to the IT department. The white paper discusses what information IT needs to have to properly create the packaged application.

The Manager – This IT staff member oversees the packaging process, determines that the necessary information is available to create the package, owns the pre-deployment process, and assigns packaging tasks.

The Repackager – This IT staff member creates the Windows Installer (.msi) package, then customizes and tests the Windows Installer installation before handing it off to the deployment process.

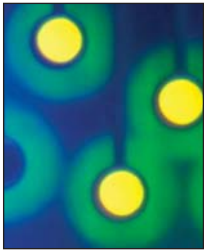
Each class of user needs accessible and secure tools to view the status of the pre-deployment process and keep the project on task. The white paper examines how InstallShield AMS™ adds functionality and significant management benefit by providing a high-level view into the pre-deployment packaging process.

by David Chernicoff
Senior Contributing Editor, *Windows & .NET Magazine*

Contents

| | |
|---|---|
| The Pre-Deployment Workflow Process | 2 |
| Why InstallShield AMS™? | 3 |
| The Requestor | 3 |
| The Manager | 4 |
| The Repackager | 5 |
| Improved Repeatability, Reliability, and Efficiency | 6 |
| About the Author | 6 |

Managing the Software Packaging Process— A Crucial Part of the Application Lifecycle



As businesses grow, so do the number of line-of-business applications and the complexity of those applications. In addition, many companies integrate office automation applications into the mix of custom business tools and customize these otherwise off-the-shelf applications to meet the needs and standards of the corporation. Whereas many businesses use application deployment tools, both of the home grown and Independent Software Vendor (ISV) variety, to get applications to the servers and desktops where they're needed, these organizations often overlook the need to properly prepare their applications for deployment.

Although a generic approach might work for preparing to deploy some software applications, one can't underestimate the cost-savings and customization benefits of properly prepackaging many of these applications. Keep in mind that just getting the applications onto the required servers and desktops isn't the end of the deployment process. Consider how much time you spend to ensure you've properly configured the applications for their intended role and that the applications have the necessary resources to operate most efficiently. By incorporating the application's final configuration into the pre-deployment planning process, you can preconfigure the application and any other necessary files to create an exceptionally cost-effective process. As such, skimming on the tools and processes to prepare applications for deployment makes little sense. This white paper defines the roles and responsibilities of the three classes of users who touch the pre-deployment process and shows you how you can obtain significant management benefit through a high-level view into the entire pre-deployment packaging workflow process—from accepting and tracking the initial request for a packaged application to the handoff to application deployment tools.

The Pre-Deployment Workflow Process

Understanding the workflow process necessary to properly prepare applications for deployment is crucial to keeping track of and reporting on the status of the pre-deployment process. Slapping together a Windows Installer (.msi) package in an ad hoc fashion might serve the needs of a small organization, but as organizations grow, the inefficiency of the ad hoc process becomes quickly apparent.

Even in a small business that uses software to deploy applications, a strong process methodology for creating Windows Installer packages for deployment is a cost- and time-saving investment. With the ad hoc model, making

Skimming on the tools and processes to prepare applications for deployment makes little sense.

a minor change to a deployment package can necessitate starting the entire packaging process from scratch. Whereas with a strong pre-deployment packaging model, minor changes are just that—minor tasks that make use of the stored information about the existing packages and that don't require starting over each time a change is requested.

Most software deployment applications are found in larger organizations that are likely deploying dozens of applications. Consider the chaos that can result without a structured methodology for packaging each of those dozens of software applications for deployment. Add in to the mix the need to upgrade applications or to customize them for use in specific business areas, and your IT department is faced with a never-ending stream of catch-as-catch-can packaging projects that are poorly documented, rarely repeatable, and hard to track.

Communicating that workflow status also has significant importance, not only to the IT staff responsible for providing the pre-packaged applications for deployment, but also to those business unit managers and users who request the applications and configurations to be prepared for deployment to their groups. When the ability to track the status of the requested pre-deployment is available to the original requestor as well as to the IT staff working on deployment projects, the chances of miscommunications are significantly reduced. This point is more important than it might sound: Poor communication

between users and IT is incredibly common and often results in projects having to be redone and additional problems that cause delays.

Why InstallShield AMS™?

InstallShield AMS is a system for managing the pre-deployment Windows Installer packaging and testing process as well as handing off these packages to your existing software deployment tools. The InstallShield AMS Web portal provides links to the integrated InstallShield AMS workflow management, customized reporting, application data storage, and application request management tools. From the point of making the formal request for an application to be packaged to the point of handing off the prepared Windows Installer package to the deployment software, InstallShield AMS gives the application management processes a much needed structure that significantly improves the repeatability, reliability, and efficiency of the entire process.

Because all interaction with InstallShield AMS takes place through a Web browser, you can submit requests for application packaging and access status reporting on existing workflows from any workstation in the enterprise. With InstallShield AMS, each corporate user who has a stake in packaging and deploying the application has a well-defined role in the application management process. Each class of user has its own needs and requirements for managing the pre-deployment process, including easily accessible and secure tools. To assist each type of user, tools providing a view into the status of the project play an important role in keeping the project on task. Detailed reporting and auditing information gives the manager the ability to identify and correct workflow bottlenecks and realize efficiencies, thus ensuring that the end-user package is completed in an accurate and time-sensitive manner.

For the purposes of this white paper, we will define the stakeholders as follows:

- The Requestor – This end user is typically a business manager who is responsible for a department and

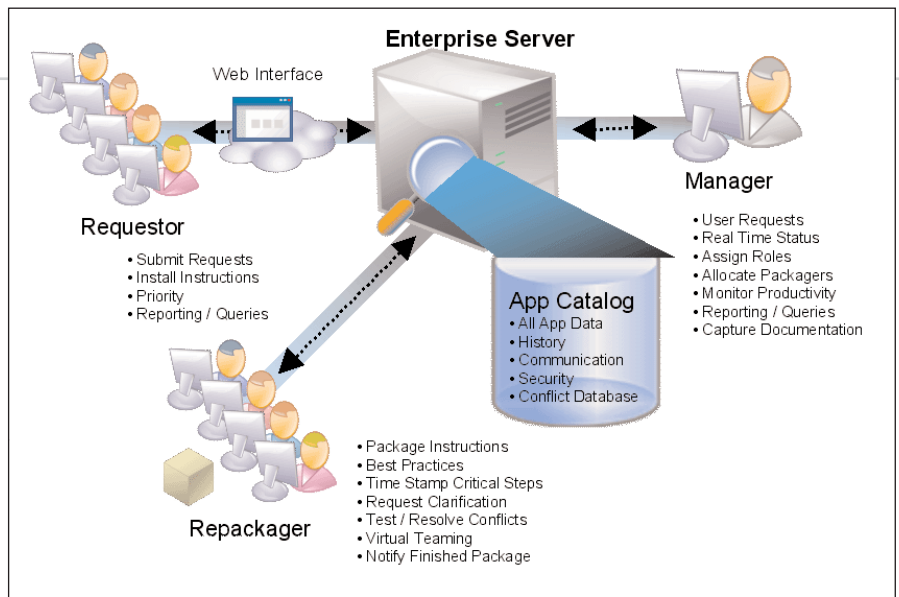


Figure 1: InstallShield AMS™ – The Application Management System

who defines the needs for a specific application and submits a request for the application to the IT department. This role can also be assigned to an IT staff member tasked with the direct responsibility for supporting a specific application or department.

- The Manager – This IT staff member oversees the packaging process, determining that the necessary information is available to his or her staff to create the proper package and owning the pre-deployment process. The manager should also be able to assign work and monitor the status of the project.
- The Repackager – This IT staff member is responsible for creating the actual Windows Installer package. The repackager creates and tests the Windows Installer installation before handing it off to the deployment process.

All these stakeholders are crucial to the successful completion of the pre-deployment process; each has predefined responsibilities to the other stakeholders. Figure 1 shows the flow of information among these roles in InstallShield AMS.

InstallShield AMS provides all of the stakeholders with real-time views of the status of any project for which they have responsibilities. InstallShield AMS also records and stores all communications that use the InstallShield AMS portal and lets you easily, and permanently, document the process, thus highlighting

potential problems and minimizing the number of false starts that can occur.

The Requestor

No one understands their business unit's application needs better than the users of those applications. Departmental managers or IT specialists who are tasked with requesting application deployments should also be responsible for providing all necessary information to the IT department for a successful packaging and deployment. The InstallShield AMS portal provides access to easily customizable templates that walk the requestor through the information he or she needs to provide so that the IT staff can provide a package that meets the requestor's needs.

The information that the requestor needs to provide in the repackaging request might seem obvious, but let's take a closer look:

General Information

- Application name – What's the name of the application? (e.g., Microsoft Visio). The requestor should also provide the version name or number (e.g., Microsoft Visio for Enterprise Architects 2003).
- Application description – What does the application do? For example, Microsoft Visio could be described as "Office Automation Software."
- Priority – How important is this software request to the respective business unit or organization?

- Specific hardware requirements – What are the hardware requirements for the computer on which the software will be installed?
- License agreement – What is the End User License Agreement (EULA) that must be agreed upon for the software to be lawfully installed?
- Support agreement information – What technical support requirements are needed for the requested software?
- Customization information – Are there child processes or custom code that must run to complete the setup? Are there different packages for a clean installation and for an upgrade installation?
- Networking or database connectivity concerns – Are there any conflicts that need to be resolved before installing this software? Does the software require ODBC or Data Source Name (DSN) set up?
- Other software dependencies – Does the software need any other software installed (e.g., Adobe Acrobat to read documentation)?

Specific Installation Instructions

- Password, serial number, or registration information – Any specific information that must be used to install the software, including registration codes and package passwords.
- OS requirements – Will the software package be installed on only Windows XP, only Windows 2000, both OSs, or other legacy OSs? Are their differing requirements based on the OS?
- Deployment method – Is the package being prepared for use by a deployment system such as Microsoft Systems Management Server (SMS), LANDesk, or IBM Tivoli?
- Installation process – Is the installation silent or does it require end-user interaction? If not silent, the requestor should define the end-user interaction.
- Uninstall process – Should the uninstall process be done silently? Are there specific screens that should be hidden from the user during the uninstallation process?

The more information that the requestor can provide up front, the greater the chance the IT staff can successfully prepare and deploy the application. With InstallShield AMS and its dynamic Web templates, the IT department can create custom templates that prompt the requestor and walk them through the entire process, ensuring that a standard set of information is received with each packaging request. For many tasks, the standard templates provided with InstallShield AMS are suitable to task; dynamic templates let the user go beyond what the standard templates offer so that you can customize the process as necessary.

The Manager

The role of the manager encompasses more than just the responsibility for

managing one packaging project. InstallShield AMS collects information about the packaging workflow process in real-time, routing the applications to the correct step in the workflow process. The manager has the ability to define the process steps for InstallShield AMS, then let InstallShield AMS adjust the process on the fly for each packaging project. As the manager gains more experience with the deployment process, that individual can fine-tune the process steps as often as necessary, and InstallShield AMS will adjust any dependencies according to the specific requirements of the software being packaged. In the end, InstallShield AMS takes unnecessary workload off the manager and handles it in software.

In this role, the manager also needs to be well informed about the overall packaging process because this person will be responsible for customizing the InstallShield AMS portal's Web templates. The manager can completely customize dynamic templates without the need to hire a Web designer to write HTML code. The manager can also continually update and modify the portal templates to set up and enforce best practices and standards for your business. In addition, the manager can modify what information InstallShield AMS captures, the process workflows, the requestor Web forms, and any Web form that the manager believes is needed to provide the process and information data required by the packaging projects.

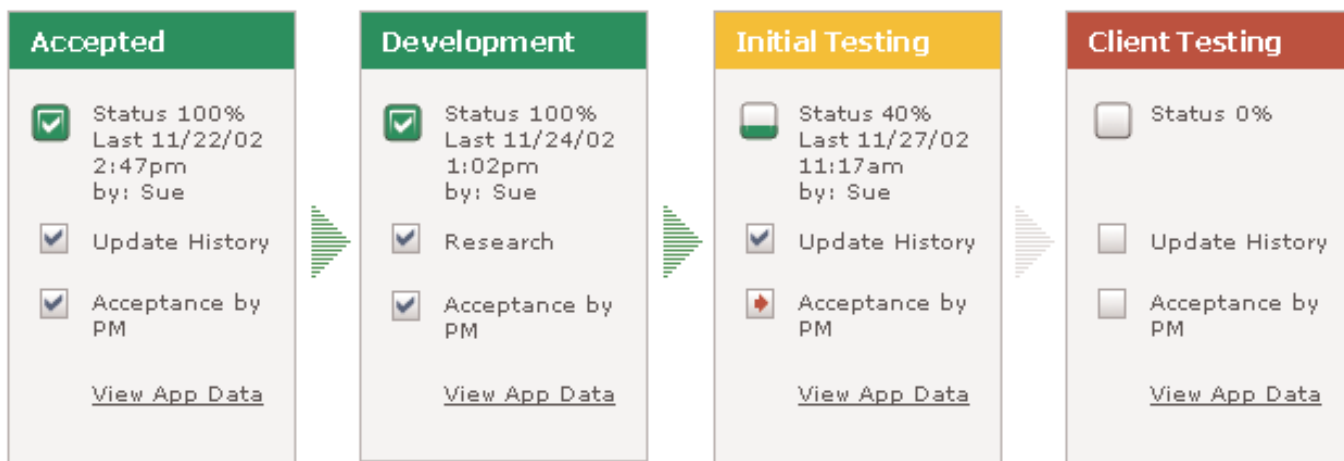


Figure 2: InstallShield AMS™ Workflow

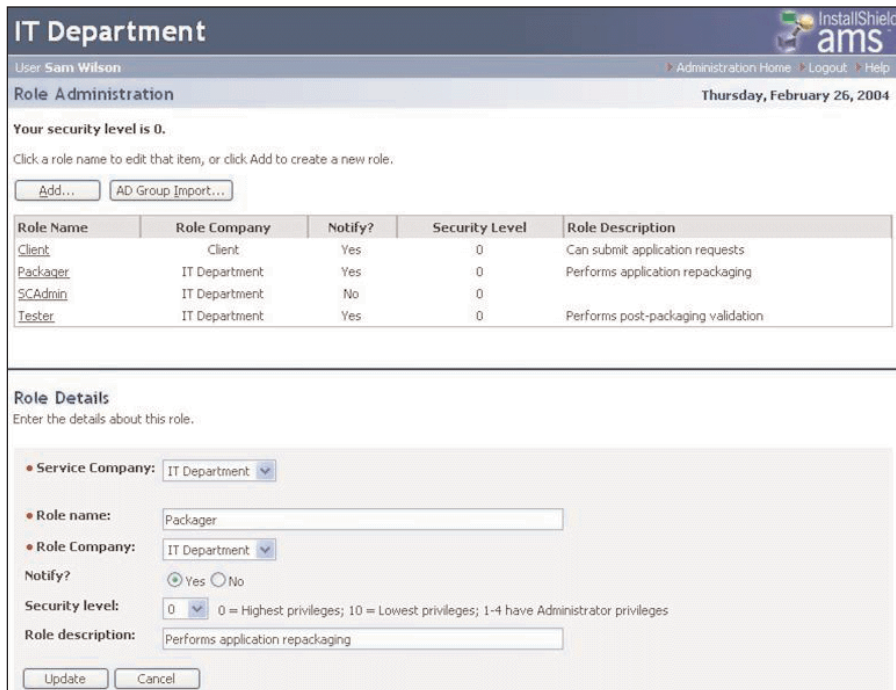


Figure 3: InstallShield AMS™ Role-Based Security Model

The InstallShield AMS portal also includes the InstallShield AMS Dashboard, which is a single Web page that the manager configures to provide the real-time status of any application being packaged. In addition to the dashboard, InstallShield AMS can send template-based emails to the stakeholders in the packaging process that automatically notify the stakeholders when milestones are reached, supply general status reports, and provide alerts to any problems that might crop up.

One of the InstallShield AMS portal Web templates lets the manager assign specific tasks to an IT staff member. InstallShield AMS gives the manager the ability to assign work through a template on both an application and specific task basis. As a result, the manager can assign a specific step in every packaging process to the same IT staff member (e.g., for quality assurance testing) or assign an entire packaging process, from start to finish, to a specific staff member. Not only can the manager make the assignment, but InstallShield AMS lets the manager monitor those assignments to track the staff member's status on a real-time basis. Monitoring also extends to the direct integration that InstallShield AMS has with Microsoft SMS,

giving the manager an end-to-end view of the application packaging and deployment process, from one console, if the delivery mechanism is SMS. This monitoring is in addition to the problem tracking and resolution facilities that InstallShield AMS offers. Figure 2 shows the InstallShield AMS workflow.

Because InstallShield AMS captures all the data related to a packaged application as well as all the communications between people working on the packaging, a detailed audit trail is always available to assist in problem resolution. InstallShield AMS also offers its own tools to facilitate problem resolution in the application packaging process, including in-process queries to be sent to the requestor who asked for the application package. This audit trail is an invaluable tool, not only for resolving problems but for seeing what works well, too.

For reporting purposes, InstallShield AMS provides the InstallShield AMS Report Wizard, which lets the manager design, store, and run custom reports on any application process that InstallShield AMS is managing. Anyone who has ever managed any sort of corporate IT project knows how important good, accurate reports are to the successful completion of any project.

InstallShield AMS and its Web-based InstallShield AMS Reporting Wizard make standardized and ad hoc reporting simple and easy to accomplish.

Last, but certainly not least, is InstallShield AMS's security component. The manager configures the complete role-based security model provided within InstallShield AMS. Every user who needs to use InstallShield AMS can have specific access rights and clearly defined privileges. The security model integrates with the Windows NT domain model or with the users defined with Microsoft SQL Server user accounts, thereby eliminating the need to manage the security model independently of your standard network security management. Figure 3 shows the InstallShield AMS role-based security model.

The Repackager

Traditionally, many companies have taken the job of preparing a software package for distribution and handed it to an IT staff member almost as an afterthought. For large corporations that have realized the importance of properly packaging and deploying software, the role of the repackager has become more important. However, tools for managing this process have typically not been readily available.

InstallShield AMS gives the repackager the opportunity to make sure that all the information necessary to perform a successful packaging and deployment project is provided up front. The auditing tools minimize miscommunication and eliminate problems that might not be discovered before the requestor can storm into the manager's office and say "This isn't what I asked for." The repackager can refer to the detailed data provided by the workflow templates and request additional information throughout the process using the tools and facilities built into InstallShield AMS, keeping the audit trail complete.

After the repackager begins testing the actual deployment and functionality of the deployed application, he or she can provide detailed reports to the

requestor and ask questions about behaviors that the requestor expects the application to perform. The manager or any user with the appropriate rights to InstallShield AMS can easily monitor this back and forth communication between the repackager and the requestor to help keep everyone in the loop and on the same page. This constant feedback and analysis lets the repackager suggest changes to the workflow process to help ensure that the best practices and most efficient processes become part of future InstallShield AMS-based projects. Figure 4 shows a sample InstallShield AMS application report.

InstallShield AMS lets the repackager contribute to deployment workflows with the ability to define and select the deployment sites for the packaged application. In addition, InstallShield AMS provides special deployment workflows that can begin after the package is ready to be staged, tested, and deployed. These features make the repackager's job that much simpler, with well-defined guidelines to ensure a smooth and trouble-free process. Ultimately, InstallShield AMS lets the repackager provide a smooth handoff of the complete, tested, packaged application to the software deployment tool that your corporation has standardized on and ensures a happy requestor who has a trouble-free software package to deploy to his or her users.

Improved Repeatability, Reliability, and Efficiency

InstallShield AMS takes the pre-deployment packaging process out of the realm of the ad hoc, do-it-as-time-permits IT task and lets you easily manage the entire process (from the request for software to the handoff to the deployment tools and possibly beyond) in an efficient manner that gets buy-in from all necessary parties who touch the project. Business units finally get to see inside the process and are treated as the integral components of the IT process that they've always been. The fact that InstallShield AMS offers a real-time view into the entire process can only raise these business units' confidence level

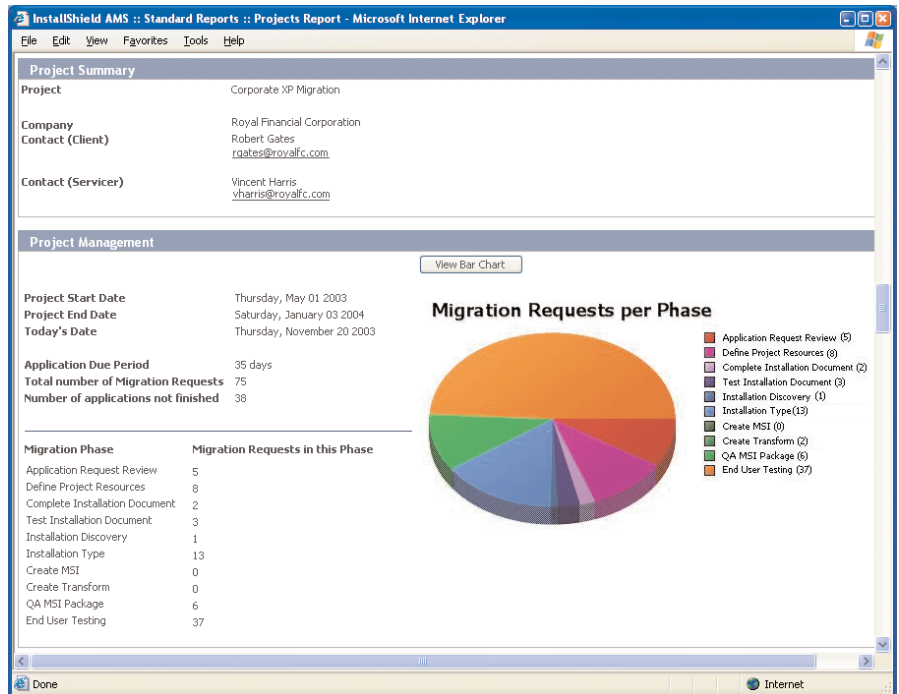


Figure 4: Sample InstallShield AMS™ Application Report

and make them feel that IT is responding to their needs in a timely and efficient manner.

InstallShield AMS provides IT staff members who are assigned the job of packaging projects with the tools they need to make sure they're on the correct path through the pre-deployment process. The detailed audit procedure gives confidence to both the IT department and the requestor that his or her needs and desires have been accurately represented, and that the project timetables are realistic and understood by all parties.

With InstallShield AMS, the manager finally gets the global view of the entire application packaging and deployment process. This real-time information, maintained by the InstallShield AMS application, keeps the manager in the loop. With just a few clicks in the Web browser, the manager can ascertain the status of every packaging project in process as well as see all historical data on completed projects.

InstallShield AMS has the unique position of keeping not only IT but also line-of-business personnel informed about the progress of their application management projects in real time. This type of information is rarely available to business users without having to directly go to IT for the answers. The

InstallShield AMS model facilitates not only the packaging and deployment of software applications but also facilitates the communicating process between all parties involved in making a successful project.

Additional features within the InstallShield AMS software are designed to tightly integrate InstallShield AMS not only with other applications, such as InstallShield AdminStudio and deployment tools such as Microsoft SMS, but also to make InstallShield AMS look like an integral part of a corporation's line-of-business applications, with custom branding and tight integration using the Web portal model. Careful and detailed consideration of best practices and the enhancement of established corporate standards make InstallShield AMS a high-value performer in your IT tools arsenal.

About the Author

David Chernicoff (david@winnetmag.com) is a senior contributing editor for *Windows & .NET Magazine*. He has been writing computer-related features and product reviews for more than 15 years and is coauthor of *Microsoft Windows XP Power Toolkit* (Microsoft Press).