Measuring the Returns of Software Packaging

A White Paper Prepared for
Wise Solutions, Inc.

ENTERPRISE MANAGEMENT
ASSOCIATES

ENTERPRISE MANAGEMENT ASSOCIATES

Measuring the Returns of Software Packaging

Table of Contents

Measuring the Returns of Software Packaging

Introduction

Of all the challenges and concerns faced by today's IT organizations, one common issue consistently tops the list: software deployment. In both large enterprises and small, the basic question of how to efficiently deliver new software to end users—efficiently, quickly and with a minimum of errors—has become one of IT's most complex challenges.

There are a number of drivers behind this heightened focus on software deployment, but the most compelling driver is simple volume. Today's corporate enterprise is using software to automate nearly every business function, including design, manufacturing, sales, distribution, purchasing, and accounting. Software is becoming an increasingly important tool not only for processes inside the enterprise, but for interfacing with customers, suppliers and other trading partners. Not only are there more applications being implemented in the enterprise than ever before, but the mission-criticality of those applications is hitting new heights.

Today's corporate enterprise is using software to automate nearly every business function, including design, manufacturing, sales, distribution, purchasing, and accounting.

Responding to these challenges—as well as their own need to boost revenue opportunities in a slow economy—software vendors and developers are rolling out new releases at an unprecedented rate. While software makers once favored large software suites and long release cycles, today's application vendors are modularizing their offerings and issuing more narrowly-focused releases at a faster rate of rollout than ever before. Security and other rapidly-changing issues also are driving many software vendors to issue patches and updates at an unprecedented rate, adding even more volume to the software deployment challenge.

End users also are adding to the software deployment agenda. As today's users become increasingly sophisticated in their use of applications and devices, they pose new challenges for the IT organization. With the emergence of Web technologies, users have become much more adept using a broad array of applications – including those that may not be approved for installation on their office desktops. Many users also are more mobile than ever before, changing their desktop locations frequently and using wireless devices that may add new challenges to software deployment. Today's software deployment processes must take into account not only a growing number of applications internally and externally, but a new range of software demands from an increasingly sophisticated user base.

How is this software expansion—from the business, from software vendors and from end users—affecting the IT organization? One immediate impact is that IT organizations are now responsible for preparing and deploying a broader diversity of software than ever before. From operating systems to billing software, from ERP tools to e-mail clients, software deployment now involves a wider spectrum of capabilities—and in some cases, customization requirements—than in past years.

IT organizations also must achieve a new level of responsiveness in their software deployment efforts. Recent security threats—including viruses, worms and newly-uncovered vulnerabilities in popular application packages—have put a new premium on the swift delivery of patches and updates to servers and end stations. In the past, most software deployment efforts focused on deploying new versions of applications, but today there are typically more patch and update deployments than new application deployments.

In the Microsoft Windows world, IT organizations also face another set of challenges: the need to upgrade the operating environment. Many enterprises still are struggling to understand the benefits and implementation issues surrounding the new Windows XP operating system, and the effort to make this upgrade brings new variables to an already-complex applications management problem. The addition of Microsoft's Windows Installer (MSI) technology to the Windows XP environment could eventually make software deployment and management much easier, but unless enterprises have an effective method for implementing MSI—and a

comprehensive plan for implementing Windows XP standards—it is likely that they will not get the full benefits of the MSI technology.

Windows Installer offers several key features for administrators and users in large enterprises. Principally, it provides a standard approach to managing the state of an application, including configuration, installation, modification, upgrade, and removal. This approach makes it easier for administrators to set and enforce corporate software configuration standards, as well as reducing the frequency of user errors in installation. Windows Installer also offers a number of other features that improve application deployment, including powerful self-repair and rollback capabilities that dramatically reduce the occurrence of deployment-related desktop software problems.

The diversity of applications, the need for customization, the frequency of updates and the complexity of the evolving Windows operating environment all are driving enterprises toward one common goal: standardization. IT organizations want to set software release and configuration standards across the enterprise, so that they no longer have to "reinvent the wheel" each time they do a deployment. Setting common specifications for software version and configuration across the enterprise can simplify IT administration significantly over time. However, setting these desktop software standards—and enforcing them through a common method of software deployment—is a complex and time-consuming task that typically cannot be achieved using out-of-the-box software.

What resources does IT have to meet these challenges? Unfortunately, the answer typically is "not enough." In a slow and difficult economy, there is a critical need to keep IT costs down while maximizing the productivity of technical staff. In the area of software deployment, most companies are seeking to deliver a larger number of software releases, updates and patches while reducing the number of staffers required to do so. Enterprises are seeking out ways to make software deployment staffers more productive, while minimizing the need for specially-skilled personnel.

In a slow and difficult economy, there is a critical need to keep IT costs down while maximizing the productivity of technical staff.

However, enterprises cannot afford to cut corners on software deployment. Each time a software deployment fails to reach a desktop—or causes a problem when it gets there—it triggers actions by service desk staffers and IT technicians who must diagnose, trace and repair the problem. While some enterprises still handle deployment by simply pushing the application to the desktop and waiting for the help desk to call, more savvy IT organizations are seeking out ways to achieve a higher rate of successful installations. These savvy IT groups understand that while it may be initially cheaper to "send and pray," it is much more cost-effective in the long run to test their software in advance and reduce expensive IT service calls.

In fact, the most enlightened IT organizations today measure their success in part by their ability to deploy software to as many desktops as possible on the first attempt. This benchmark is important not only as a measure of IT productivity, but of employee productivity in the enterprise. If software deployments fail on their initial distribution, they prevent employees from taking advantage of new applications designed to improve productivity. Worse, an untested software deployment may actually cause end user machines to operate improperly, rendering employees helpless until an IT technician can fix the error. By deploying software correctly, and transparently, on the first try, IT organizations can boost overall business productivity.

With so much pressure on IT organizations to keep costs down—and to improve IT and employee productivity in the process—many enterprises have sought out new tools to aid in software deployment. Until recently, these tools tended to focus on the distribution side of software deployment – the process of pushing ready applications out to the desktop. Over the last decade, many large Windows shops have made a significant investment in software distribution tools such as Microsoft's Systems Management Server (SMS)

and/or third-party distribution tools from companies such as Novell, Altiris, Novadigm, Marimba and a host of others. In many enterprises, these tools have helped IT to evolve from the old days of "sneakernet"—in which software was physically delivered and manually installed on each desktop—to a cleaner, more efficient network-based mode of automated software distribution.

Today, however, most large enterprises understand that software distribution is only part of a broader deployment process. To ensure swift, successful deployment of software—and to guarantee that the software meets the needs and skills of the end user as well as the business—many enterprises are turning to emerging technology for software *packaging*. Software packaging tools help IT organizations to analyze new applications; edit and customize applications to meet specific business or end user needs; eliminate conflicts that might occur between the new applications and the existing desktop configuration; test the applications to ensure that they will work on the desktop; and prepare the applications for swift deployment. In the following sections, we provide a description of the benefits of software packaging tools, based on industry research as well as interviews with users of the leading software packaging tool, Wise Solutions Inc.'s Wise Package Studio.

To ensure swift, successful deployment of software—and to guarantee that the software meets the needs and skills of the end user as well as the business—many enterprises are turning to emerging technology for software packaging.

The Value Proposition For Software Packaging

Exactly what are software packaging tools and what are their benefits? In a nutshell, software packaging tools help to automate the processes that occur between the time an application is delivered to the IT organization—either from an internal development team or from an independent software vendor—and the time the application is distributed to user desktops. Software packaging tools take the "pray" out of "send and pray," enabling the IT organization to analyze, prepare and test applications thoroughly to ensure that they will work immediately upon receipt by the end user.

There are many products on the market that claim to offer packaging capabilities, but not all of them are complete packaging toolsets. To be a true packaging solution, a toolset must include seven basic capabilities:

- **Analysis.** A software packaging tool or toolset helps IT staffers—sometimes called packaging specialists—to evaluate a new application and determine its compatibility with existing desktop software configurations. The tool may also help the packaging specialist to identify customization requirements or other tasks necessary to adapt the software to a given group of desktops and/or users.
- **Capture.** A software packaging tool or toolset provides the ability to automatically collect all necessary information about an application and its installation procedures so that the application can be repackaged and installed on client devices. While most packaging tools take a simple "snapshot" of the application, others provide more sophisticated capture capabilities that enable packaging specialists to gather a wider range of dynamic information and do more sophisticated software packaging.
- **Customization.** A software packaging tool or toolset enables the packaging specialist to modify the software to meet corporate standards, as well as to fit the specific requirements of the end users or desktops that will receive it. This process may include adding enterprise-specific interfaces or features as well as changing the way the software will be configured in order to better comply with corporate standards.
- **Validation.** A software packaging tool or toolset enables the packaging specialist to validate an application against pre-established guidelines or policies to ensure compliance. Such guidelines might include operating system specifications, internal corporate specifications or industry standard specifications.

- **Conflict resolution.** A software packaging tool or toolset enables the packaging specialist to identify potential conflicts between the new application and the existing desktop software configuration. It should also help the packaging specialist to resolve those conflicts by automating the replacement of potential problem components with components that are consistent with the existing desktop configuration.
- **System testing.** A software packaging tool or toolset provides the means and/or methodology to conduct tests on lab machines that emulate production desktops to ensure that any conflicts or other installation problems are resolved in a test environment prior to broad distribution.
- **Integration with distribution tools.** A software packaging tool or toolset provides a direct interface to automated software distribution applications, such as Microsoft SMS, to ensure that the software is distributed exactly as it was packaged.

Although many tools on the market offer some of these capabilities, very few of them offer all seven. By requiring all seven of these features in a request for proposal, IT organizations can quickly narrow their packaging tool search to just a few vendors.

Key Benefits

What benefits can an enterprise realistically expect to achieve through software packaging tools? The answer will differ from enterprise to enterprise, depending on the maturity and sophistication of its existing processes and technology for software deployment. However, at some level, a packaging tool or toolset should deliver benefits in four core areas:

- **Greater reliability of applications.** One of the most important benefits of a software packaging solution is its ability to detect and correct potential conflicts between existing applications and applications that are scheduled to be deployed. By using packaging tools first, enterprises can significantly reduce their rate of installation failure and make applications much more predictable. On the IT side, packaging tools can cut support costs dramatically by identifying conflicts that historically have taken weeks—even months—to locate, and by reducing troubleshooting and user downtime following deployment. Finally, packaging tools can help enterprises to take broad advantage of Microsoft Windows Installer, which also helps make applications more reliable from the first boot through features such as application self-healing.
- **Increased speed of deployment.** In the early days of desktop computing, enterprises installed shrink-wrapped software on every workstation. Today, however, software vendors and internal software developers offer a wide range of options and customization capabilities that can make the difference between efficiency and inefficiency on the desktop. Packaging tools enable enterprises to explore all of those options, quickly identify the configuration best suited for the user, and then pass a fully-tested package to the software distribution system. By preparing software for each desktop in advance – and by eliminating errors that might otherwise occur during deployment – packaging tools help to increase the speed and success of the deployment process.
- **Improved software standardization.** By implementing enterprise-class packaging tools and processes, enterprises often find that they gain an opportunity to bring order to their desktop applications chaos. With packaging tools, enterprises can prescribe and enforce an exact configuration for each Windows application, down to the Data Link Library (DLL) level. Through this process, the IT organization can swiftly reduce the number of versions of each application that it must support, while enforcing standard configurations that make desktops easier to support. The packaging process can help companies tailor

software capabilities to the needs of specific groups of users while helping to prevent users from making flawed or unauthorized installations.

- **Institution of best practices**. On the IT side, the implementation of packaging technology presents a golden opportunity to create more efficient best practices for analyzing, testing and deploying new applications. In fact, to achieve the full benefit of packaging tools, enterprises should consider implementation of several key best practices, including:
  - *Universal deployment of Windows Installer.* Microsoft's Windows Installer technology could significantly improve the ease with which applications are added to the desktop, and could lower support costs by eliminating errors associated with those installations. However, most enterprises have not deployed Windows Installer across all of their applications, and therefore can gain only some of the technology's benefits. Packaging tools can help enterprises add Windows Installer capabilities swiftly across their spectrum of applications.
  - *Institution of a conflict management process.* Even when IT organizations today identify conflicts between old and new applications, they do not always have a structured method for resolving the conflicts. Packaging tools can help to quickly establish common conflicts, and then aid in establishing policies for conflict resolution as well as automating the process.
  - *Centralization of the packaging process.* Although many Windows applications are used across the enterprise, many enterprise IT organizations do not have a common organization and/or methodology for software packaging and deployment. A packaging toolset can facilitate this centralization, providing common training, methodology and tools that work across business units, geographies, or other organizational boundaries. This centralization eliminates redundancy, improves training efficiency, and speeds implementation of new tools or packaging processes across the enterprise.
  - *Structured application grouping.* One of the most common problems with application deployment is the diversity of software used within a large enterprise. While some applications are used across the organization, others may be useful only to a business unit, geographic location or a group of specialized users. By instituting a packaging process, IT organizations can establish a structure for classifying and handling applications that are core to the business, applications used primarily for departments or business units, and applications that are to be deployed ad hoc to small groups of users.
  - *Formal application request and approval process.* The institution of packaging technology is a golden opportunity for enterprises to establish a formalized process for requesting, approving and distributing applications. Through a simple workflow process, enterprises can provide the means for users to request a specific application and get approval from the appropriate managers on both the business and IT sides. This process could significantly aid in eliminating unnecessary application deployments, while also ensuring that end users receive the appropriate, pre-defined versions of the requested application, thus cutting the cost and complexity of support.

Purpose and Objectives of the ROI Study

EMA recently undertook an ROI study on software packaging to help IT executives and staff to better understand the potential operational and financial benefits of the technology. During the course of this study, EMA collected and analyzed data from a base of enterprises currently using Wise Solutions Inc.'s Wise Package Studio solution. This study was conducted to identify the cost-related and productivity-related variables associated with software packaging tools. Respondents were questioned on their operational practices, efficiencies gained and investment made in software packaging technology, both before and after the implementation of Wise Package Studio. The study was undertaken in order to:

- Identify the key benefits of software packaging tools, specifically Wise Package Studio.
- Identify the costs associated with licensing and implementation of these tools.
- Quantify how a company's business can be improved through the implementation of software packaging tools and processes.
- Determine the return on investment achieved by major enterprises using Package Studio, and the potential ROI for prospective Package Studio users.

Key Findings

Analysis of the data from the Wise Package Studio customer base yielded several important findings. Among the most significant:

- Payback on the software packaging investment was rapid – and in most cases, extraordinarily short. Several users paid for their entire Package Studio investment—including training and consulting costs—with a single month of IT resource and time savings achieved through the technology.
- Enterprises achieved significantly shorter total packaging times with Package Studio, even if they had already been using older Wise packaging tools. Each user's experience was different – some reduced capture time, others reduced validation time, others reduced editing/customization time, and still others reported shorter testing time. All of the enterprises reported heavy savings in at least two phases of the packaging process.
- Virtually all of the respondents achieved major cost/time savings in the area of conflict resolution. Several respondents said they reduced the time required to identify and resolve conflicts from days—even weeks—to a matter of hours.
- All of the respondents reported a lower failure rate in their software deployments after implementing Package Studio than before. This increased success rate generally translated to higher user productivity within the line of business.
- All of the respondents reported a lower incidence of service desk calls following software deployment. This lower trouble report rate generally translated to reduced costs at both the help desk and field service levels.

Virtually all of the respondents also reported some impact from process and resource improvements that were not possible to quantify in the data. For example, most respondents said the software packaging process helped their organizations to take better advantage of Microsoft Windows Installer; many others said the packaging process has helped their organizations to build more stable, enforceable standards for desktop configuration.

EMA's Return On Investment Process

EMA uses a three-step process for its ROI studies. During this process, end users are interviewed and operational practices are reviewed.

Step 1 – Collecting the data points. EMA analysts first collect data through a series of conversations with end users. These conversations use a survey technique to provide information about implementation, operations, efficiencies and business impact.

Step 2 – Analysis. Gleaning comparative and quantifiable results from the data is the second step. EMA has developed several proven methods for data analysis. EMA analysts also use their experience to understand the business and technical context of the responses, and factor those variables into the analysis.

Through the analysis, EMA derives quantifiable data on the benefits of a given technology – both from a business perspective and from an IT perspective – as well as costs, both obvious and hidden.

Step 3 – Calculating the ROI. Using the collected data points from Step 1 and the analysis from Step 2, EMA analysts then calculate the ROI by first normalizing the inputs for time periods and personnel numbers, then grouping the results into two basic categories:

- Benefits include operations savings, process savings, improved performance/speed, decreased error/failure rates, faster and/or less expensive maintenance/recovery.
- Costs include software license costs, staff time costs, training costs, consulting costs, and lost IT productivity costs

All benefits and costs that can be quantified are factored into the equation to achieve an ROI for each specific user. For purposes of this paper, we have also reported on the average ROI achieved among the study respondent base.

Where to Find ROI in Software Packaging Technology

Although the bulk of the software packaging process takes place prior to deployment, enterprises that implemented Wise Package Studio found that the technology had an impact on their deployment and post-deployment processes as well. To gauge the full ROI of the technology, EMA interviewed enterprises about their experiences in all three phases of deployment and exacted both objective and subjective data on the costs and benefits of packaging technology. The following three sections provide capsule results on enterprise experiences in each phase of the software deployment process.

Sample User 1
Regional Utility Company

Total target desktops: 5,000

Average number of applications deployed per month: 7

Average number of targets per deployment: 1,200

- *Chief pre-deployment benefit of software packaging:* Reduced total packaging time by approximately 100 man-hours per application.
- *Chief post-deployment benefit of software packaging:* Reduced service desk calls by 71 percent – an annual savings of more than $600,000 in service desk time alone.
- *Total ROI* (ratio of benefit to cost in one year): 32:1

Pre-Deployment Phase

Analysis. The process of evaluating a new application and determining the need for customization or modification is generally a manual one. Although Wise Package Studio users reported very little real time or cost savings during this initial phase of deployment, several of them said that the packaging process added structure to the analysis phase that did not previously exist. For example, several companies used data collected from previous packaging experiences to craft templates for evaluating new applications and exposing potential obstacles or deployment challenges while the software was still in the selection or development process.

Capture. For years, the process of "capturing an install" has relied on simple technologies that take a "snapshot" of an application and collect all of the data required for replicating it and distributing it to enterprise desktops. Wise Package Studio offers a new level of sophistication to the capture process, not only collecting more detailed data, but also offering more dynamic capture capabilities that do not depend entirely on a single snapshot in time. Several enterprises in the study reported that Package Studio enabled their organizations to cut capture time by 50 to 75 percent. Companies that already had strong capture technology in place—such as older Wise software—saw a less radical reduction in capture time, usually on the order of 5 to 25 percent.

Editing/customization. As stated earlier in this paper, there is a growing trend toward tailoring applications —even software modules or patches—to fit the specific needs of a group of users. Thus, virtually all of the

enterprises interviewed are stepping up their customization efforts, even for shrink-wrapped software that previously might have been distributed without modification. The majority of Wise Package Studio users saw a reduction in editing/customization time following the implementation of the Wise solution. A large bank reported that Package Studio enabled it to reduce customization time by more than 90 percent, from more than a week to just a few hours. Other users reported cuts of 50 to 66 percent in customization time and resources; these results were much more typical.

Validation. The process of validating applications against pre-established guidelines or specifications is a new one to many large enterprises. In some cases, the introduction of packaging tools is the instigator for the initiation of a validation process – in fact one major insurance company reported that Package Studio increased, rather than reduced, validation time because the company had virtually no process for validation prior to the introduction of the Wise software. Of those enterprises that previously had done validation—the majority of the study respondents—virtually all reported that Package Studio had reduced the time and resource requirements. A large bank said that Package Studio cut its validation time by some 98 percent, from approximately 8 business days to about one hour. Other enterprises reported more typical reductions in validation resources of around 50 percent.

Sample User 2
Major Financial Services Company

Total target desktops: 32,000

Average number of applications deployed per month: 10

Average number of targets per deployment: 2,000

- *Chief pre-deployment benefit of software packaging:* Reduced conflict resolution effort from approximately 40 man-hours per application deployed to approximately 1 man-hour per application deployed.
- *Chief post-deployment benefit of software packaging:* Reduced average deployment failure rate by 7 percent – a savings of more than $2 million per year in service desk and maintenance time.
- *Total ROI* (ratio of benefit to cost in one year): 23:1

Conflict resolution. Of all the processes supported and/or automated via Package Studio, conflict resolution was, by far and away, the most frequently-cited area of benefit. Prior to installing Package Studio, many respondents relayed war stories about "DLL hell," in which new Windows applications contained multiple DLLs that conflicted with existing applications and caused installation failures, system lockups, or even workstation crashes that required multiple system rebuilds. The only way to identify the conflicts was to test virtually every function of every application prior to rollout – and even when a problem was exposed, it was often difficult to correctly identify the source of the conflict. Many of the study respondents reported spending weeks, even months, in the conflict resolution phase for a single application.

Because Package Studio is one of a very few solutions that automates the conflict resolution process, none of the study respondents had an automated tool for this function prior to implementing the Wise product. As a result, most of the enterprises reported radical, dramatic improvements in conflict resolution time following the implementation of Package Studio. A major chemical company stated that Package Studio enabled it to cut conflict resolution times from approximately 8 weeks per application to about 1 week – a savings of nearly $9 million annually. A major financial services company reported that it cut its conflict resolution effort from about one week per application to about one hour. A bank reduced its conflict resolution time from 20 hours per application to about 15 minutes. In several instances, enterprises achieved full ROI for Package Studio through conflict resolution savings alone. In other words, that one feature paid for all of the costs associated with Package Studio—including implementation and training costs—in a matter of months.

Acceptance testing. In order to ensure success of deployment, most enterprises conduct a series of acceptance tests on a small number of desktops before releasing an application for full distribution. Typically,

this process entails testing the application on a set of laboratory PCs configured to look like typical target desktops, or on a limited number of production machines. Then the application is usually tested on a small subset of the deployment base, usually "power users" who are willing to participate in the testing in order to get the applications first. In most cases, enterprises reported that Package Studio helped to structure the applications for the test environments, but very few of the respondents reported any time savings as a result of using packaging tools during the acceptance testing process. Most of the respondents said that acceptance testing remains one of the longest processes in software packaging, simply because it must be done by end users over a period of weeks in order to ensure reliable feedback. One bank did report that Package Studio enabled it to cut acceptance time by over 60 percent, from approximately 8 business days to about two and a half days.

Sample User 3
Fortune 500 Chemical Company

Total target desktops: 52,000

Average number of applications deployed per month: 100

Average number of targets per deployment: 200

- *Chief pre-deployment benefit of software packaging:* Reduced conflict resolution effort from approximately 8 weeks per application deployed to approximately 1 week per application deployed – a savings of nearly $9 million a year.
- *Chief post-deployment benefit of software packaging:* Reduced average deployment failure from approximately 5 percent to less than 1 percent.
- *Total ROI* (ratio of benefit to cost in one year): 15:1

Improved exploitation of Microsoft Windows Installer technology. Although the results were impossible to quantify, virtually every enterprise in the study reported that the use of Package Studio had a direct impact on the implementation and use of Microsoft's latest standard for software installation: Windows Installer (MSI). This technology, which is supported natively by Windows 2000 and XP, often is not fully exploited because it requires that applications be packaged in the MSI format prior to installation, a requirement that is currently met by relatively few software vendors. The addition of a sophisticated packaging tool such as Package Studio allows companies to migrate legacy, non-MSI compliant software to the Windows Installer format, essentially unlocking the full potential of Windows Installer and enabling enterprises to use the Microsoft technology to better manage software and add reliability to the application deployment process.

One major utility company estimated that Package Studio helped it to reduce the time to migrate applications to the Windows Installer format by approximately 60 percent. Other responses were more subjective, but all of them indicated that Package Studio had a major impact on the deployment, installation and use of the Microsoft Windows Installer technology.

Total packaging time. Enterprises in the study varied widely in their evaluation of the returns offered by Wise Package Studio in different functional areas. While some users found heavy savings in capture time or validation, for example, others reported that Package Studio had no significant impact on those functions. No matter what their responses in specific functional areas, however, virtually all of them reported significant savings in the total packaging process, from analysis to testing. In general, most enterprises reduced packaging time by about 50 percent – a few users reported total packaging time savings of as little as 25 to 30 percent, but several others reported savings of well over 50 percent. Every user surveyed saw at least a 25 percent savings in packaging time and resources per application.

Deployment Phase

Although Wise Package Studio is not a software distribution tool and does not directly deploy software, most of the enterprises surveyed did see a direct relationship between institution of the packaging solution and a greater efficiency of software deployment. Some enterprises reported reductions in deployment time of 60 to 75 percent following the implementation of Package Studio. Others said that their deployment

times have not changed, but the tight integration of software packaging tools with software distribution tools has led to a more automated process that puts less pressure on IT staff.

Streamlined handoff between development and distribution. In the case of Wise Package Studio, many enterprises have been able to take advantage of an electronic "handoff" that moves applications from the packaging phase directly into a software distribution tool. This approach smooths the workflow between software development staff and the software distribution staff, paving the way for faster deployment to the end user. Essentially, the packaging solution provides a bridge between software development and software distribution that did not exist before.

Sample User 4
Large Regional Bank

Total target desktops: 2,500

Average number of applications deployed per month: 11

Average number of targets per deployment: 400

- *Chief pre-deployment benefit of software packaging:* Reduced validation effort from approximately 60 man-hours per application deployed to approximately 1 man-hour per application deployed.
- *Chief post-deployment benefit of software packaging:* Reduced average deployment failure rate by 14 percent – an annual savings of more than $800,000 in service desk and maintenance time.
- *Total ROI* (ratio of benefit to cost in one year): 42:1

Higher rate of deployment success. Even more importantly, every single enterprise surveyed has recognized a higher rate of successful deployment since the introduction of Package Studio to their environment. This means that for each application deployed, organizations saw a lower percentage of desktops which failed to install the new software. The size of the improvement varies – one bank cut its deployment failure rate by 14 percent, while an insurance company eliminated only 1 percent of failures. But in every single case, at least a small increase in successful deployments was realized.

Enterprises attributed much of their improved success to Package Studio's conflict resolution capability, which enabled them to identify and prevent potential deployment problems. Many of these conflicts previously went undetected until they cropped up on the user's desktop following a major deployment, but such situations are much less likely to occur when software packaging tools are in place, the enterprises said.

By eliminating failed deployments, many IT executives believe, the packaging solution also improved end user productivity. In the past, end users who experienced deployment failure also experienced other unwanted consequences, such as system lockup or even hard drive failure. Since the deployments typically are remote, failures often require the user to become involved in the deployment, from making configuration changes with a service staffer via telephone to simply standing aside while IT staffers find and correct the failure. Package Studio reduces deployment failure rates, thus reducing the frequency of the end user's involvement in troubleshooting and keeping non-IT staffers in "work" mode, rather than "wait" mode.

Post-Deployment Phase

Software packaging technology is most directly used during the pre-deployment and deployment phases, but its effects are felt after deployment as well. In the ROI study, enterprises reported packaging benefits in several post-deployment areas:

Reduced calls to the service desk. As stated previously, one side effect of the "send and pray" method of software distribution was a high incidence of help desk calls following a software deployment. These calls generally occur because new software has an unanticipated impact on the behavior of the end user's desktop system—from a missing icon to complete system lockup—and the end user calls the service desk to inquire about the problem. In the best case, these calls use up the time of help desk staff, incurring unintended costs

on the enterprise. In the worst case, these calls lead to IT maintenance efforts that might take skilled IT staffers days or weeks to resolve.

In the study, nearly every respondent reported that the implementation of Wise Package Studio resulted in a reduction in help desk calls. Most of the enterprises said that the reductions were modest—generally between 1 and 12 percent—but that the overall impact on IT service and maintenance over the course of a year was significant. In one extreme, a utility company reported that it was receiving service desk calls after approximately 80 percent of its software deployments. With the implementation of software packaging technology, the company has reduced that figure to about 8 percent – a benefit of more than $600,000 in service desk staff time alone.

Sample User 5
Major Insurance Company

Total target desktops: 8,100

Average number of applications deployed per month: 15

Average number of targets per deployment: 75

- *Chief pre-deployment benefit of software packaging:* Reduced total packaging time by approximately 60 man-hours per application – a savings of more than $200,000 a year.
- *Chief post-deployment benefit of software packaging:* Reduced average deployment time from approximately 20 man-hours per application to approximately 4 hours per application.
- *Total ROI* (ratio of benefit to cost in one year): 3:1

Faster first-tier problem identification. Prior to deployment of Package Studio, many enterprises reported that their first-tier service desk representatives often had difficulty identifying the source of the desktop problems reported by end users following deployment. A number of enterprises reported that service desk personnel were spending as long as five hours attempting to identify a single user's post-deployment problem – the industry average for service desk calls is 15 to 30 minutes. With packaging technology in place, however, deployment-related problems generally are easier to identify, since the sources of application conflicts can be brought to the surface using conflict resolution and testing technology. One utility company reported that it cut the length of the average post-deployment service desk call by 50 percent following the introduction of Package Studio.

Fewer help desk calls escalated to higher service tiers. If a first-tier service desk representative cannot identify or resolve a user's trouble report, he/she will then "escalate" the call to a more skilled IT staffer for evaluation and/or repair. These higher-tier calls are more expensive to the enterprise than the first-tier calls, because they require more highly-trained personnel and a more sophisticated level of diagnostic tools. A major chemical company reported that the incidence of higher-tier service calls was reduced by 20 percent following the implementation of Package Studio. Other enterprises said they also recognized some benefits of packaging technology at the higher service tiers, although the overall workload of their IT staff did not change during the year.

Less time spent on diagnostics at higher service tiers. When calls reach the higher service tiers, most IT organizations expect that the nature of the problem will be clearly identified, if not resolved. In software deployment, however, problems are often caused by complex conflicts between interdependent applications, and these conflicts can be difficult for even the most skilled staff to identify. Several enterprises reported that with the availability of conflict resolution and testing features delivered by Package Studio, higher-tier service technicians spent less time in problem diagnostics when handling post-deployment trouble reports.

Less time spent on problem repair at higher service tiers. Once the nature of a post-deployment application problem has been diagnosed, the service technician must quickly go about the task of repairing it. Without packaging technology, this repair time could be extensive, because there was no structured method for handling it – for example, most enterprises had not identified a protocol for choosing the "correct" DLL among a group of two or more conflicting DLLs. With the introduction of packaging technology, however, several enterprises reported swifter repair times at the higher tiers.

Lower incidence of system rebuilds. When higher-tier IT technicians cannot identify or repair an application problem—or when such a repair is deemed too time-consuming to be worth the effort—many technicians will resort to a complete rebuild of the desktop in order to restore the workstation to its pre-deployment state. Such a process may require hours of remote service time, or it may require a visit to the desktop by a field technician. Between the time required to travel to the problem site and the lost productivity incurred by users left temporarily without a workstation, these system rebuilds are among the most expensive consequences of a poorly-packaged software deployment. In the study, a number of enterprises reported that they had reduced the number of system rebuilds due to deployment problems by an order of magnitude. Other enterprises said they also have seen the impact of packaging technology on system rebuilds, though the overall frequency of rebuilds had not decreased over the last year.

Higher user productivity. The true measure of any IT management or provisioning tool is its impact on end users. In almost every case, Wise Package Studio users reported a lower incidence of end user downtime due to software deployment errors than they experienced without Package Studio. One bank, which was experiencing end user productivity problems in a gaudy 12 percent of software deployments, reduced its incidence of downtime to 2 percent. Several other respondents reported cuts of 1 to 2 percent, reducing deployment-related user downtime to near zero. Clearly, the packaging solution delivered significant returns not only to the IT departments studied, but directly to the business units as well.

Conclusions

Although a great deal of complexity has grown up over the concept of ROI in recent years, the basic formula has not changed: benefits divided by costs. In the case of Wise Package Studio, the benefits are so strong—and the costs so relatively low—that the majority of enterprises studied achieved full ROI within the first month of implementing the software packaging technology. Let's look at some of the results to gain some insight into these impressive numbers.

Benefits

As we saw in the previous section, Wise Package Studio has the potential to deliver strong benefits in virtually every phase of pre-deployment, from package analysis to acceptance testing. The software packaging solution also holds significant potential to reduce costs and improve efficiency during deployment and in the post-deployment phases, speeding software distribution and ensuring a low frequency of service and maintenance following deployment.

In the EMA study, no user reported total benefits of less than $400,000 during the first year of implementation. This is remarkable, given the fact that not all of the interview subjects were Fortune 1000 companies – in fact, some of the respondents were medium-sized enterprises. The high rate of benefit—regardless of company size—suggests that software packaging may be even more valuable, relatively speaking, to smaller companies than to the largest enterprises. Given the low license cost of the packaging technology, a medium-sized business might achieve bottom-line gains via Package Studio without a heavy up-front investment.

Among the enterprises studied, the most significant benefits were generally derived from improved conflict resolution, reduced packaging time, higher success in first-time software deployment, and lower cost of service/maintenance. Almost every enterprise experienced at least marginal gains in each of these areas after deploying Wise Package Studio, and every enterprise made a major leap in at least one or two of these areas. From the data collected, it is logical to conclude that prospective Package Studio users could also expect to improve their environments in each of these four areas, and make a major leap forward in at least one.

Costs

Although EMA made a thorough effort to identify any "hidden costs" associated with packaging technology in general and Wise Package Studio in particular, enterprises generally reported very few unexpected costs. Most of the companies studied already had a packaging team (usually two to four IT staffers) in place, and therefore did not incur any new staffing costs. In fact, most of the respondents said the software helped make the packaging team more productive, and they therefore had more staff resource time available. Cost impact on IT resource availability was negligible, but this ROI study does factor in the lost productivity of IT staffers who stopped working in order to attend Package Studio training courses.

According to the enterprises surveyed, there are four principal costs associated with implementation of Package Studio: license fees, training, consulting and equipment. Let's take a brief look at each.

License fees. In general, the cost of software was reported to be the highest expense in implementing Package Studio. For large and medium-sized customers, Wise offers two editions of the solution: Enterprise Edition, which offers enterprise application integration capabilities and a structured, collaborative approach to software packaging for decentralized teams; and Professional Edition, which offers packaging and conflict resolution capabilities targeted at centralized workgroups. Most of the enterprises surveyed had installed both editions. The number of licenses installed ranged from three to 100, depending on the size of the packaging team. Most of the enterprises averaged less than $10,000 per license purchased.

Training. Enterprises differed in their approach to Package Studio training. While some organizations limited their training to the individuals who would be directly involved with packaging, others required related groups—such as those responsible for software distribution—to take the training as well. One company even asked upper-level executives to attend the training, in order to provide a better understanding of the packaging process. Regardless of the size and scope of the training, however, none of the enterprises studied spent more than $40,000 for training, and several respondents spent less than $5,000.

Consulting. Although Wise Solutions does provide professional services, several organizations chose to bring in third-party consultants to help with the implementation of the packaging process. In most cases, these were enterprises that needed consulting for a broader project, such as Windows XP migration or implementation of a structured packaging process. While the majority of respondents spent no money on consulting, a few companies reported spending between $20,000 and $50,000 for consulting services during their Package Studio implementations.

Equipment. The software packaging process generally requires at least one server-class Windows workstation, and most of the enterprises surveyed were packaging software on multiple machines. Virtually all of the respondents reported equipment costs in the $8,000 to $10,000 range. A few enterprises were able to handle the packaging task by using existing hardware, in some cases even putting other applications on the packaging workstations.

ROI

The return on investment for each enterprise's Package Studio implementation was calculated by totaling the real-dollar benefits of the technology—time savings, improved productivity, higher deployment success rate, and other benefits—and dividing them by costs. The results are expressed in ratios for the first year of implementation – a 1:1 ROI, for example, would mean that the technology paid for itself in full during the course of the first year.

The results of the Package Studio ROI study are impressive. The majority of respondents achieved ROI of more than 10:1 during their first year, and a few enterprises reported ROI of more than 30:1. The poorest

ROI reported was a large insurance company, which achieved 3:1 ROI in its first year of using Package Studio.

These results are even more impressive in light of the fact that some benefits of Package Studio could not be quantified. For example, virtually every respondent stated that Package Studio made a major positive impact on its implementation of Microsoft Windows Installer technologies, but because these benefits could not be quantified, they are not included in the numbers given above. Similarly, there was no way to quantify the impact of the structured packaging process on the effort to achieve enterprise software/configuration standardization, another commonly-cited benefit of Package Studio.

The process of implementing software packaging technology—and the question of which areas/functions provide the greatest benefit—may be a matter of some debate between enterprises. However, the overall value of the technology cannot be questioned. With a relatively small initial investment, enterprises can expect to reap major cost savings and improved productivity within the first year. And the technology will pay for itself within the first six months of use. Any software packaging request for proposal—indeed any software packaging initiative—must include Wise Package Studio as a primary alternative.

About Wise Solutions, Inc.

Wise Solutions is the leading provider of application installation and management software and services to Fortune 1000 companies.

Corporate Headquarters
47911 Halyard
Plymouth, MI 48170

Phone: 734.456.2100
Web: www.wise.com

About Enterprise Management Associates, Inc.

Enterprise Management Associates, Inc. is the fastest growing analyst firm focused on the management software and services market. EMA brings strategic insights to both vendors and IT professionals seeking to leverage areas of growth across e-business, network, systems and application management. Enterprise Management's vision and insights draw from its ongoing research and the perspectives of an experienced team with diverse, real-world backgrounds in the IT, service provider, ISV and publishing communities.

ENTERPRISE MANAGEMENT
ASSOCIATES

Corporate Headquarters
2108 55th Street
Suite 110
Boulder, CO 80301

Phone: 303.543.9500
Fax: 303.543.7687
Email: info@enterprisemanagement.com
Web: www.enterprisemanagement.com

668.040703