

Realtime
publishers

The Essentials Series

Enhancing IT Administration with Run Book Automation

sponsored by

 **RES** Software

by Greg Shields

Article 1: Understanding Run Book Automation for Windows	1
Why Run Book Automation for Windows?.....	2
Why Is RBA Useful?.....	3
RBA Tools Enable Comprehensive Reporting.....	4
RBA Assists Enterprises. RBA Assists Small Environments.....	4
Article 2: The Power of Automated Systems Administration	5
Modules = Ready-Made Automation	5
Integrated CMDB = Reusability & Reporting.....	7
Automating Systems Administration = Greater Control	8
Article 3: The Power of Automated User Administration.....	9
The Peril of Non-Automation	10
RBA’s Modular Approach Works with Users.....	11
RBA Enables a Least-Privilege Approach.....	12
The RBA Approach Saves Time with User Administration	12

Copyright Statement

© 2008 Realtime Publishers, Inc. All rights reserved. This site contains materials that have been created, developed, or commissioned by, and published with the permission of, Realtime Publishers, Inc. (the "Materials") and this site and any such Materials are protected by international copyright and trademark laws.

THE MATERIALS ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. The Materials are subject to change without notice and do not represent a commitment on the part of Realtime Publishers, Inc or its web site sponsors. In no event shall Realtime Publishers, Inc. or its web site sponsors be held liable for technical or editorial errors or omissions contained in the Materials, including without limitation, for any direct, indirect, incidental, special, exemplary or consequential damages whatsoever resulting from the use of any information contained in the Materials.

The Materials (including but not limited to the text, images, audio, and/or video) may not be copied, reproduced, republished, uploaded, posted, transmitted, or distributed in any way, in whole or in part, except that one copy may be downloaded for your personal, non-commercial use on a single computer. In connection with such use, you may not modify or obscure any copyright or other proprietary notice.

The Materials may contain trademarks, services marks and logos that are the property of third parties. You are not permitted to use these trademarks, services marks or logos without prior written consent of such third parties.

Realtime Publishers and the Realtime Publishers logo are registered in the US Patent & Trademark Office. All other product or service names are the property of their respective owners.

If you have any questions about these terms, or if you would like information about licensing materials from Realtime Publishers, please contact us via e-mail at info@realtimepublishers.com.

Article 1: Understanding Run Book Automation for Windows

Think for a minute about the tasks you're asked to accomplish every day in your IT environment. You may install a printer driver or reset its Spooler service. You might add a user account or update the rights and privileges assigned to it. You might need to build an entirely new computer or reconfigure an existing one due to changes in your business needs.

Tasks such as these are at the core of Windows systems administration. The constant need for enacting change to our IT environments is precisely why the job of systems administrator or systems engineer exists. Yet, in many organizations, all of these changes are completed using manual and unrepeatable steps. Installing a printer driver manually involves logging into the server's console, locating the driver, launching its installation routine, and answering any questions that may arise. Completing this task on one computer takes exactly as long as completing it on a second. Repeating it across dozens or hundreds of computers means manually redoing these steps over and over again, a process that doesn't scale, wastes time, and ultimately reduces your effectiveness as an administrator.

As an example, printer driver installation is but one of hundreds or thousands of tasks that needn't be completed in this manner. Automated processes exist for accomplishing these tasks. But the problem many administrators have with automation is in the expense of initially coding and testing those automated processes. Spending a few hours coding an automated printer driver installation makes sense when there are thousands of drivers to install. But when there are only ten or a dozen, there is not enough benefit associated with the high cost of building that automation from scratch.

The same holds true with configuring new servers as they join the domain. There are a lot of time-intensive tasks involved with completing the initial configuration on a new server, yet those tasks are repeated every time a new server is built. Figure 1 shows how wrapping modules associated with each action into a Run Book provides a way to automate the process and eliminate the wasted time.

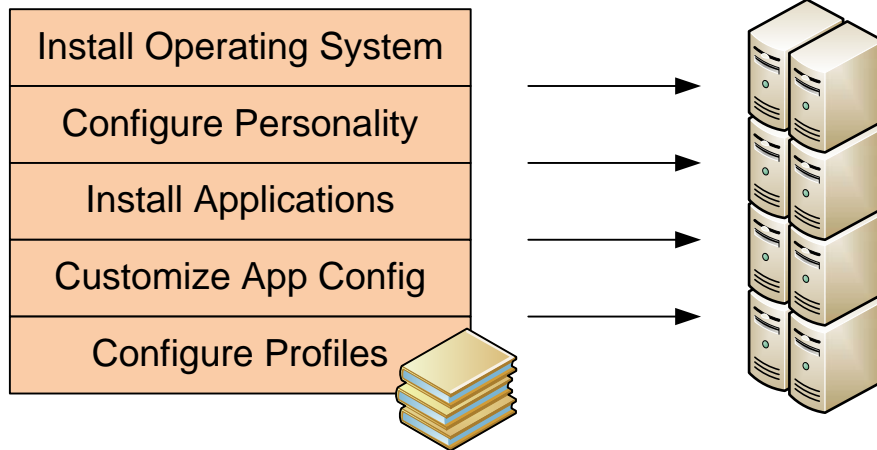


Figure 1: When multiple actions are wrapped into a Run Book, that Run Book can be repeated across any number of computers.

Why Run Book Automation for Windows?

It is for this exact reason that tools for Run Book Automation (RBA) exist. Small and medium environments are similar to enterprises in that they can benefit from the automation of common tasks. But the creation of the automation itself is the most cumbersome step. For many environments, the time and challenge in developing custom automation from scratch is usually greater than the benefit gained by automating.

Tools that leverage RBA can eliminate almost all of that initial automation creation. Through the use of pre-created modules for common actions, they provide an easy-to-use mechanism for automating common tasks in the IT environment. Because there is little to no custom coding, these modules eliminate the costly initial steps of developing the automation itself. With an effective RBA tool in place, an IT environment of any size can quickly and easily kick off a pre-generated action directly within the tool across any number of computers. It is further possible to link multiple actions together into a “Run Book” for completing multiple actions all at once.

For example, consider the common situation in which a registry key needs to be updated across every server in the data center. Updating that registry key involves using tools such as the native Registry Editor to manually connect to each remote registry and update the key. Doing so for 10 servers involves making 10 connections and updating the key in 10 separate locations. One automated option is to author a script to accomplish the same task. Yet the authoring of that script can consume hours of time in its development, testing, and ultimate deployment.

Using an RBA tool to accomplish this step involves little more than invoking its already-available “update the registry” action against the set of computers. Taking this automation to another level, if updates like this occur on a regular basis and are part of a set of needed changes, effective RBA tools consolidate changes into a single job that can be later invoked or scheduled across entire swathes of computers at once.

Why Is RBA Useful?

The implications of this level of immediate automation are clear. However, RBA provides benefits to the IT environment in other ways as well. Consider the following IT goals that can be fulfilled by running common user and systems administration actions through an effective RBA tool:

- *Repeatable.* If you use a standard GUI console to accomplish an action, the only way to perform that action again on another computer is by repeating its steps in the other server’s GUI. Actions that are constructed out of modules within an RBA tool automatically create Run Books that can be repeated over and over again. Invoking the action within the RBA tool is equally as simple as doing it within the server’s native GUI, so there is little time cost of initially creating the automation.
- *Recordable.* Actions in the IT environment that are completed through the native GUI are gone once they have completed. Yet when those actions are first constructed within an RBA tool, they can be automatically recorded into that tool’s database. After you use the tool to build a Run Book of individual actions, that Run Book can be later invoked again and again to accomplish those changes elsewhere. If the later invocation is slightly different than the first, you need only update or customize the settings to the individual action instead of re-writing and testing the entire automation.
- *Trackable.* Every action or Run Book in an RBA tool is invoked from within the tool, so each action is then logged to its secured audit database for tracking purposes. Knowing which administrator accomplished what action is a boon to troubleshooting and goes far in fulfilling the auditing requirements of industry and regulatory compliance.
- *Predictable.* Unlike manual GUI configurations, which can be prone to error as an administrator repeats them on each updated computer, actions run within an RBA tool are exactly the same every time. Once a Run Book of desired actions is created, repeating its invocation across any number of computers assures exact results. This predictability reduces the risk of incorrect changes that can lead to system downtime or other problems down the road.

-
- *Delegatable.* Optimizing the workflow of the systems administrator is a key value-add with RBA tools. Once high-level administrators identify and build Run Books out of desired actions, the invocation of those Run Books can be delegated to down-level technicians or even the Help desk. This frees high-level administrators for other more important and valuable tasks. This delegation capability in combination with RBA's predictability enables IT organizations to assign resources where they can make the most impact.
 - *Business-aligned.* Lastly, RBA and the tools provide ways to further align IT processes with business processes. This happens through the automated and scripted collection of quantitative data that assists with the recognition of service level or regulatory fulfillment.

RBA Tools Enable Comprehensive Reporting

That last bullet point is of particular use in environments that must meet and report on Service Level Agreements (SLAs) or the auditing needs of IT compliance. Gathering the right level of quantitative data to prove that IT has fulfilled its service level obligations is a painful and regular activity. Proving the configurations on computers remain the same and have not been modified is yet another arduous task. The completion of these tasks on a regular basis requires time and effort that takes away resources from value-added activities. RBA tools and their internal reporting capabilities provide ways to validate how configurations have been made. Their mapping of people to actions further proves to auditors and security personnel which administrator completed what change.

RBA Assists Enterprises. RBA Assists Small Environments.

Numerous tools exist that provide this level of functionality for enterprise environments, those with hundreds of servers and thousands of workstations. But well-designed platforms that arrive equipped with common actions and easy-to-use Run Book authoring capabilities also make sense for smaller environments. Scalability to fit the needs of both is key. The right tools provide a way to quickly and easily build a Run Book of server configuration updates or user provisioning steps without the hassle and time cost of custom script authoring. Whether invoking these actions across one, ten, or a thousand machines at once, the necessary steps for the administrator are the same.

Now that you understand what RBA is, the next step is in recognizing the types of actions that can potentially be automated using its tools. The second and third articles in this series will focus on the specific benefits small and medium-sized environments can gain through automating their systems and user administration processes.

Article 2: The Power of Automated Systems Administration

The first article of this series explained the high-level concepts of Run Book Administration (RBA). This idea involves the linking of common actions performed against computers and users in an IT environment into Run Books. Once created, the invocation of a Run Book can be consistently repeated across any number of elements in the IT infrastructure.

Completing common administrative actions within an RBA tool provides a secure and delegatable mechanism for quickly automating the otherwise tedious process of managing IT assets.

Systems administration is one half of that need. Systems administrators are charged every day with monitoring the behaviors of servers and workstations in the environment and enacting change when necessary. That change can take a number of different forms, with administrators building new servers one day while applying patches or updating configurations the next. Using traditional methods, completing these tasks involves working with the computer's console itself, directly interacting with its GUI consoles to complete any changes. Yet while these consoles are excellent for accomplishing these tasks on a single machine, they don't scale well to doing so across more than one.

Modules = Ready-Made Automation

RBA and the tools that enable it take the common actions that make up a systems administrator's workday and encapsulate them into "modules." Invoking a module within the RBA tool's interface and against a computer will accomplish the action as defined by the module. For example, a module may exist to create a new shared folder. Invoking that module against a computer will create a shared folder with any necessary configurations as defined in the module. If the module requires custom elements to properly invoke, those elements will be defined within the job.

When many administrators think of systems automation, they first think of scripting. Scripting and scripts enable automation because their encoded actions can be repeated by re-running the script. Yet scripts can be challenging to write and time-consuming to write well. As Figure 1 shows, invoking an individual RBA module is a lot like invoking a script. However, that module is presented to the administrator as an object rather than individual lines of code.

With RBA modules, all the necessary code for accomplishing the module's actions is run in the background without the need for complicated programming. All the administrator must do is invoke the module and answer any questions it requires to complete the action. Built-in to the RBA platform are methods of invoking that action against identified systems in the environment and linking multiple modules and their actions together to create a Run Book.

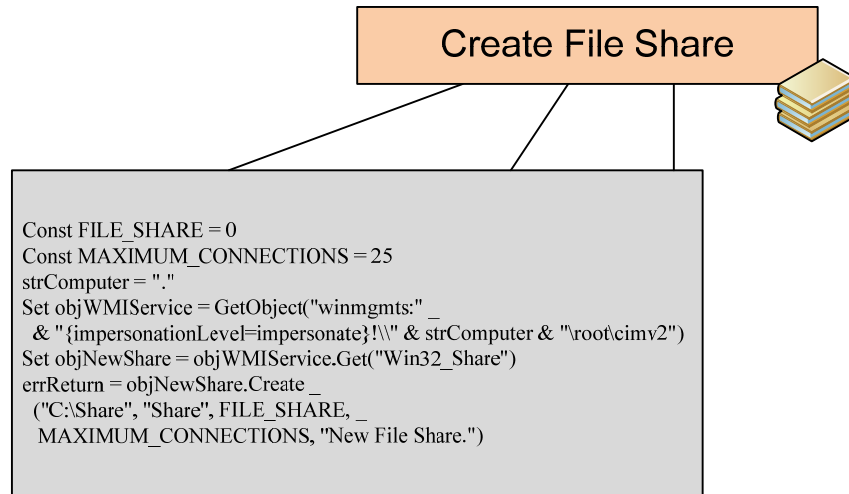


Figure 1: RBA platforms expose actions in their interface. Underneath those actions is the pre-created code that accomplishes the action on target computers.

Different RBA platforms will expose different types of modules in their interfaces. A best-in-class platform should include the modules you require for the administration of the systems in your environment. It should also include easy-to-use authoring tools for linking modules into Run Books. Consider some of the following modules as a starting point for the Run Books you might need to create.

- Creating shared folders
- Creating home directories, profiles, or other folders for users
- Pre-populating profiles with configurations
- Imaging systems along with personalization capabilities
- Snapshotting system configurations to enable configuration versioning and rollback
- Applying patches on computers
- Installing software onto desktops, laptops, and servers
- Querying servers for configurations such as installed software, hardware, registry settings, file and folder permissions, and disk space
- Adding printer drivers
- Consolidating event logs, pulling event logs into a centralized database where only a limited number of users can view and modify data

With modules such as these available for orchestrating operational changes across computers, it is easy to see how virtually every component of systems configuration can be automated. Going a step further, with a little work in building the right Run Book, a candidate system can be taken from bare metal to fully configured through a few clicks of the mouse. This rapid deployment and rapid configuration capability brings levels of agility never before seen to environments both big and small.

Windows Administration Needn't Be Domain-Centric

The Windows domain is a powerful tool for consolidating users and computers into a single entity for authentication and resource authorization. Yet the boundary of the Windows domain can add administrative complexity for environments that have more than one domain to manage.

RBA tools enact change based on agnostic modules that are customized as they are built-in to Run Books, so it is advantageous for IT organizations to leverage RBA platforms that work across domains. This is particularly the case in development, testing, and staging environments where multiple domains are common. When considering an RBA platform, look for those that can support the automated configuration change of computers irrespective of their domain location.

Integrated CMDB = Reusability & Reporting

Many environments leverage multiple point-solution tools for accomplishing their daily administrative tasks. One product may provide imaging support for new computers. Another may deploy and enforce configurations based on established policies. A third and fourth may be responsible for automatically deploying software and applications. These point solutions automate their tasks of responsibility, but they do so with distributed databases and separate consoles that do not integrate with each other.

An excellent example of this limitation is in determining what is actually installed to a particular desktop system. If you use one mechanism for deploying OS patches and a second for installing applications, each mechanism works with its own database. Information about what is installed to desktops is stored in two places, making difficult or impossible the creation of an inventory report. However, RBA platforms that incorporate the capabilities of both mechanisms log their changes to a central Configuration Management Database (CMDB). Pulling a report on a system's installed patches and applications can be done through the platform's console interface.

This integrated CMDB gains greater utility when you consider the modules and Run Books that have been applied to a particular system. The configuration of modules and Run Books are usually stored within the RBA platform's database, and any invocation is logged for later auditing. Figure 2 shows how this might work. In Figure 2, the integrated CMDB associated with the RBA tool provides history information about the computer \\server1. Any Run Books that have been invoked against \\server1 are listed in order. Because the configuration of each Run Book itself is similarly logged in the database, each and every individual change to that computer can be tracked. With additional information about the date and time the action was invoked as well as the user who invoked the action, it is easy to track the source of configuration changes. All this data enables a holistic situational awareness of systems and their configurations over time.

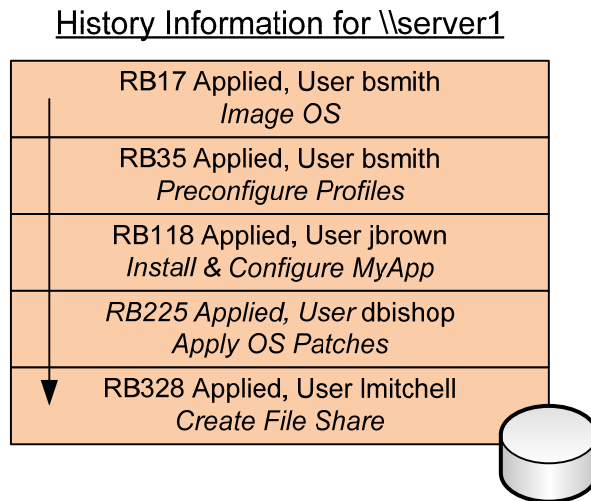


Figure 2: An integrated CMDB enables the storage of rich history information about changes made to a computer.

Automating Systems Administration = Greater Control

Managing a computing environment is a challenging job. With thousands of possible configurations on each of dozens or hundreds of computers, getting your arms around all of them is impossible without the right tools in place. Running common administrative actions for systems through an RBA platform reduces duplicated work, improves the application of changes across all systems, and enables centralized logging and auditing for keeping an eye on the history of your environment. RBA and its tools enable these capabilities without complex and time-intensive custom coding and with centralized CMDB storage of environment configurations.

Yet systems administration is not the only area in which RBA can assist the Windows administrator. The complex management of users and their privileges is another area in which RBA tools excel. With the right orchestrations in place, RBA can take the pain out of user administration by automating virtually every facet. The next article in this series will explain how.

Article 3: The Power of Automated User Administration

The last article in this series discussed how Run Book Automation (RBA) and its modules can be strung together into Run Books for the automated administration of systems. This method of administering isn't necessarily new. Tools of various forms and with varying capabilities have been long available, even if they don't necessarily follow RBA's module-based approach.

Where RBA can provide a unique benefit to the IT organization is in how its modules can also enhance the activities associated with user administration. User administration has long been a pain point in many organizations. Because of the historical complexities of managing and provisioning users, many organizations—even those with otherwise high levels of process maturity—still today accomplish their necessary administration through manual methods.

And yet working with Windows' and other applications' native consoles for user administration remains a complex and time-consuming task even for the smallest of environments. A major problem is in finding other management interfaces that limit the scope of user accounts to be worked with while retaining the right level of access for the administrator. Coding such an interface is entirely possible using scripting and development tools. However, the steps to do so are challenging while also ensuring proper privileges are exposed and securing against others that are inappropriate.

The Peril of Non-Automation

This lack of automation in user administration introduces the potential for costly error, incorrect provisioning, re-work, and even security issues into the environment. Granting the wrong entitlement to the wrong user can incur one or more of these problems in the environment, which adds to the potential for inappropriate exposure of sensitive information. Let's look for a minute at each of these possibilities and the possible negative side effects associated with not automating their task completion:

- *Error and incorrect provisioning.* Adding automation to user provisioning means that user accounts when created are done so with the correct entitlements at the very beginning. This reduces the chance that a user may inappropriately be given accesses they shouldn't or not be given accesses they should. In each of these cases, error in the provisioning process leads to the follow-on problem discussed in the next bullet.
- *Re-work.* Organizations that do not have automated processes in place for account management suffer under the weight of iterative permissions requests. Users in a Windows environment do not have an easy way to view the permissions they are assigned. Thus, once assigned, users must "feel their way around" to identify if they can access their needed resources. If they don't have the necessary access, users must return to the Help desk to try again. This cyclical re-work impacts the users' productivity while reducing the effectiveness of the Help desk. In environments that require the assignment of privileges across multiple applications, such as Exchange mailboxes or assigned virtual machines, re-work becomes an ever-greater drag on productivity.
- *Security issues.* In the most egregious of cases, the incorrect assignment of privileges to users can be a security risk all its own. When incorrect privileges that enable access to sensitive data are accidentally handed out, the organization will experience a security event. These events have ramifications that are personal, organizational, political, and budgetary.

RBA's Modular Approach Works with Users

RBA's modular approach to user administration enables automation for many common user administration tasks. As with systems administration, RBA's modules can be quickly invoked or linked into Run Books for the rich orchestration of user administration needs. Actions that can be a component of an RBA platform's suite of user administration may include:

- Creating user accounts
- Creating accounts for applications such as email
- Provisioning accounts and entitlements
- Deprovisioning
- Assigning applications, both local and remote
- Creating profiles based on user role and requirements
- Enabling workflow associated with user administration
- Allowing non-technical departments such as Human Resources (HR) to accomplish tasks of user administration

With the capabilities associated with these modules in mind, consider the Run Book orchestration seen in Figure 1 that can be built within the RBA platform. There, the user's request drives the creation of their user account, mail account, assigned applications, and profile. In Figure 1, the first step is expanded further to show the internal steps that can be configured within that module. As with RBA's impact on systems administration, this codification of this process means that user creation events occur exactly the same every time. This reduces error, eliminates re-work, and reduces the chance for a security event.

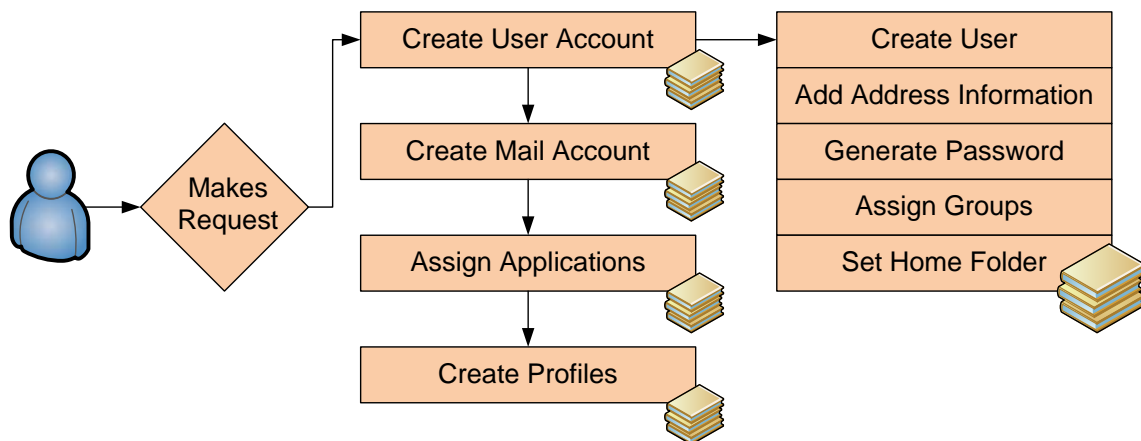


Figure 1: Run Books related to user account creation events can involve multiple and tiered activities, all of which are encapsulated in the RBA platform.

RBA Enables a Least-Privilege Approach

Where RBA and its tools provide specific benefit to this end is in the ability to tailor which components of user administration are delegated to which individuals. The concept of Least Privilege requires that administrators are given exactly the level of privileges that are required to accomplish their jobs. This is difficult to do with native Windows tools, as administrator rights are often required to accomplish necessary administrative tasks such as those involved with users. However, layering an RBA platform atop existing Windows and other application interfaces means that the RBA platform's internal permissions structures can be used in enabling the correct levels of access for each administrator.

The bulleted list in the previous section takes this idea a step further by suggesting that even non-technical users such as those from HR can assist with the user administration process. With the right types of Run Books in place for user creation, the proper distribution of consoles for invoking those Run Books, and the correct levels of permissions, IT can relocate the operational burden of user administration to others in the organization. Those individuals, such as those from HR, may have an improved capacity to ensure users get the right accounts as they arrive on their first day of work. They can also ensure that accounts and entitlements for users are revoked as they leave their positions with the company.

The RBA Approach Saves Time with User Administration

RBA's repeatable and module-based approach to user administration ensures the cohesive application of your business' policies. It further provides a mechanism to offload elements of the user administration workload to alternative groups that may be better suited to perform those actions. Using an RBA platform that combines user administration with systems administration, comprehensive Run Books can be created that link user administration with systems provisioning, reducing the overall cost to manage the users in your environment.