

# Seven Steps to a Smooth Migration with Terminal Emulation Software

Rapidly advancing technologies and increasing pressure for better return on investment are constantly driving changes to desktop software. Terminal emulation software is one of the most deployed desktop solutions in the IT world, connecting countless users to legacy green screen applications running on mainframes and AS/400s. This is why any change to the terminal emulation solution, whether it is upgrading from an older version or standardizing terminal emulation, must not be regarded lightly. As an industry expert in the terminal emulation market, OpenText has developed a Change Management Cycle methodology that has successfully helped organizations replace more than half a million seats of terminal emulation software.

Table of Contents

Introduction.....3

Winds of Change.....3

Striking the Right Balance.....4

Seven Steps to Successful Software Deployment .....4

    Listing your requirements.....4

    Evaluating offers .....4

    Recreating the environment.....5

    Migration and conversion .....5

    Security .....5

    Deployment.....5

    Support and maintenance.....5

Final Thoughts .....6

About OpenText .....6

## Introduction

Change is good—if it is controlled. In the IT world, where ever-evolving technology and the pressure to deliver return on investment are always present, it's not unusual for a little change to become a chaotic endeavor. On the list of potential chaos-generating IT projects are those related to the replacement of desktop software, which can affect the work of many, if not all employees, in an organization. Terminal emulation software falls into this category as it is often deployed to hundreds or thousands of machines throughout an organization and around the world.

The main function of terminal emulation software is to offer the same functionality on a PC as the original hardware terminal. Organizations that use terminal emulation software typically have legacy green screen applications installed on mainframes, AS/400s, or UNIX machines. These applications are critical to running their businesses, and the cost of replacing those applications outweighs the potential gain that such a replacement could bring. Even today, tens of millions of individuals around the world access a green screen terminal interface to accomplish their work.

## Winds of Change

There are many reasons a company would consider changing its terminal emulation software, and all of them eventually come down to ROI. These are the high-level drivers that lead to this decision:

**Technology:** Some triggers that compel organizations to review software solutions include migration from SNA to IP-based communication, a change in operating systems (Microsoft® Windows® 7 to Windows 10, for instance), or a shift in the IT paradigm from desktop to thin client computing. Often software solutions must be updated to remain compatible with the new platform or infrastructure.

**Economics:** Strategic choices, such as acquisitions and mergers, organizational realignments, and cost reduction plans, often translate into changes at the tactical level. These types of changes cause departments to reconsider their budgeting decisions and, therefore, their software purchases.

**Vendors:** Not all terminal emulation vendors are created equal. Some vendors may have poor customer service, unfriendly channels, or a lack of focus in this area of their business. Combined with declining revenue, these issues have put a lot of pressure on some vendors over the last 10 years and have led customers to reconsider their choices.

**Regulations:** Government and industry regulations, such as SOX, HIPAA, Gramm-Leach-Bliley, and Basel II, as well as industry standards such as strong SSL/TLS encryption (FIPS 140-2), Smart Cards (FIPS 201), and PCI Data Security, have prompted businesses around the world to perform thorough assessments of their IT infrastructure (including software and hardware decisions) in order to reach the required compliance levels.

## HIGHLIGHTS OF THIS PAPER

- *Review reasons for changes in the terminal emulation market*
- *Identify stakeholders in a terminal emulation standardization project*
- *Propose a straightforward seven-step methodology to manage the project*



## Striking the Right Balance

Software replacement projects are often handled by two types of stakeholders in an organization: economic buyers and technical buyers. Both parties must collaborate effectively by managing the different aspects of the project to ensure a successful outcome. While economic buyers are worried about reducing direct and indirect costs, technical buyers are motivated by ease of deployment and impact on project deadlines.

Cost of acquisition, a high priority for the economic buyer, is not a sufficient indicator of which vendor should be chosen. Instead, an effective selection strategy consists of collaborating with the technical buyer to determine the time required to migrate from the previous product, as well as the ongoing resources needed to support the software in the future, and integrating these costs into the ROI and TCO calculations.

Besides verifying a vendor's reputation and track record in the market, economic buyers should also consider the services the vendor can offer in managing licenses. A flexible license policy and a guaranteed level of service are often overlooked but remain critical pieces in the overall success of a terminal emulation replacement project.

Technical buyers, on the other hand, usually have different priorities, starting with the need to meet deadlines and manage the constant pressure of the many projects they're handling. When it comes to terminal emulation software, the issues are complicated by the fact that many technical employees are not mainframe experts. They've grown up in a personal computer world and are often desktop support managers and not legacy system specialists. For these employees, the top priority may be simplicity and ease of deployment. Key technical aspects that should be considered include deployment methods, macro conversion, legacy API support, infrastructure impact, and security requirements.

## Seven Steps to Successful Software Deployment

This section details the OpenText Change Management Cycle, an easy seven-step methodology that should allow most organizations to identify the scope of their terminal emulation replacement projects and manage them successfully.

### Listing your requirements

At this stage, you should start building expectations by establishing goals, setting up cost and resource limits, and defining measurable criteria that will help you monitor the success of the project. Running an internal audit and working with in-house emulation "experts" will also prevent a lot of future headaches. Putting a few stakes in the ground may sound simple, but let's face it, few organizations do it. However, if they realized how powerful it is to gather a few people in a room and ask them what they need, they would never skip this vital step.

### Evaluating offers

The next step is to formalize the call for submissions. Calling your reseller and asking for evaluation copies of every terminal emulation solution available is not enough; you need to document your findings from day one. Perform some online due diligence and gather a multi-disciplinary team to review your findings. It is a good idea to involve those who will use, deploy, support, and pay for the software. Start a quick technical check by installing the evaluation copy and performing a couple of basic tasks, such as connecting to your host, navigating inside applications, and calling a few function keys.



## Recreating the environment

This step is a critical phase of your replacement project. It may seem obvious but whenever you change desktop software, you need to ensure the new product offers, at minimum, the same functionalities as the old one. Begin by checking basic emulation settings, such as protocols, keyboards, colors, terminal settings, printing, and file transfer. Then organize user groups and ask the vendor's system engineers to help you architect the environment. Always favor products that allow you to heavily customize user interface elements, such as toolbars, menus, and context menus.

## Migration and conversion

As the years have passed, your users have not only accumulated habits with the previous terminal emulator, they've also customized it to their personal tastes—colors and keyboards for the less adventurous and macros for the power users. Since you can't manually recreate every single user preference manually, look for conversion tools that automate the migration of user profiles and macros. Some of these tools not only convert macros automatically but also offer the same development environment as their competitors. Pay extra attention to APIs and third-party applications that can use your emulator as a gateway to mainframes. Not many terminal emulation solutions offer strong compatibility between the various HLLAPI, EHLLAPI, and WINHLLAPI flavors that are on the market.

## Security

Security is often considered synonymous with encryption in the world of terminal emulation. Encryption is certainly a key element of security, and every organization replacing their terminal emulation software should be aware of the risks of running clear-text protocols on their network. A simple way to mitigate this risk is to ensure that your next emulation solution supports authentication and encryption standards such as SSL, Secure Shell, or Kerberos. However, security also depends on your ability to operate the software in a locked-down desktop environment and customize the product to limit user access to necessary features.

## Deployment

Deployment is the last milestone before the software hits the user's desktop. It is extremely important to understand that application delivery is a detail, not a product choice. You may have a group of users who need a feature-rich desktop solution and another group who need to access green screen applications through a web portal, but it does not mean you need multiple software packages or multiple license types to satisfy your requirements.

You should favor solutions that allow you to give both traditional desktop and thin-client users access to the same software, with the same power and support for the same set of APIs. It shouldn't matter whether the terminal emulation session is delivered via CD or a web browser, users should still be able to run the same macro, use the same toolbar, and have the same user experience.

This type of solution gives companies the ultimate control over deciding what features their users need, and let the software take care of the deployment details.

## Support and maintenance

Now that you've successfully audited, evaluated, re-created, converted, secured, and deployed your solution, you will want to make sure your decision and effort stand the test of time—over the life of the product. This is where technical support and patches are key. When it comes to technical support, try to obtain customer testimonials. This should be easy to do and guarantees that the vendor you choose pays attention to customer satisfaction. On the patching front, make sure that patches come in two fashions: hot fixes for emergency situations, and update packs, which cumulate all previous fixes. Finally, make sure your maintenance contract includes access to the latest version.

## Final Thoughts

Changing terminal emulation software is not rocket science, but it requires a fair amount of planning to do it correctly. There are many terminal emulation vendors, but few can stand up to all the levels of scrutiny described in this methodology. If you are using terminal emulation software in your organization, take a few minutes to call your finance department and find out how much it costs per year. Many organizations take this software for granted and some vendors exploit that situation, overcharging customers for maintenance and providing them with lousy service. Make sure you're not one of these organizations.

## About OpenText

OpenText enables the digital world, creating a better way for organizations to work with information, on premises or in the cloud. For more information about OpenText (NASDAQ: OTEX, TSX: OTC) visit [opentext.com](http://opentext.com).

### Connect with us:

- [OpenText CEO Mark Barrenechea's blog](#)
- [Twitter](#) | [LinkedIn](#) | [Facebook](#)