WHITE PAPER

FOCUS ON:
APPLICATION OUTPUT MODERNIZATION

JANUARY 2009

Table of Contents

Executive Summary

In most large organizations, enterprise applications provide the data processing functions that support core business processes. Industry analysts estimate that there are between 100 and 200 billion lines of COBOL code running in active applications. Legacy COBOL and CICS programs support an estimated 70 percent of all business transactions.

Faced with ever-changing business and technology environments, organizations are looking to modernize their enterprise applications and the business-critical documents they create. Solutions from enterprise software vendors Micro Focus and Levi, Ray & Shoup, Inc. (LRS) help companies take advantage of new computing and document technologies while retaining the benefits of their proven legacy systems.

This report will outline the needs, challenges, and benefits of application- and output modernization projects. It will also detail how LRS and Micro Focus solutions work together to streamline the process of developing new high-performance systems that leverage investments in time-proven code and existing documents.

Modernization Needs, Benefits, and Challenges

For decades, the largest organizations in the world have relied on their enterprise applications for the majority of their information processing needs. Running on mainframe, UNIX, or a variety of other hardware platforms, these workhorse systems have been custom-tailored over time to support the specific requirements of businesses and their customers. Individual applications can streamline core business processes; taken together, they can give organizations a decisive edge over the competition.

The benefits of these systems come at a price, however. For example, the more a given application takes advantage of hardware- or OS-specific features and facilities, the more difficult future migration becomes. As a result, company executives and IT managers are faced with a variety of unattractive options:

- Continue running the existing systems on what are perceived to be expensive, obsolete, or non-strategic computing platforms
- Assume the time and monetary costs of custom-developing replacement systems on new, less expensive platforms
- Replace legacy systems with packaged applications that support the new platform and/or technology, in hopes that the cost savings will outweigh the training expenses and functionality loss vis-à-vis the legacy system

The same issues apply when it comes to output systems. Many documents used in a company's everyday operations are custom-tailored to meet specific business, legal, and/or regulatory requirements. In some instances, these valuable documents—and the systems that generated them—were originally state-of-the-art but were never updated to leverage more modern technologies. In other cases, a patchwork of documents and systems may have been implemented in reaction to individual needs, not as a part of a strategic effort.

The result of this tactical *"ready-fire-aim"* approach is a proliferation of data streams, device types, and hardware standards. In some cases, legacy systems required complex custom application code in order to provide special page formatting or data based on the document contents. This often involved adding special commands (e.g., printer escape sequences) into the data that could only be interpreted by a specific mainframe production printer or specialty output device.

Such environments present a variety of challenges for application modernization efforts. Firstly, the logic required to scan report data and customize document contents is typically quite complex. This is especially true for legacy applications that did not take advantage of Advanced Function Presentation (AFP) or other document composition technologies.

Perhaps more importantly, application modernization projects are often part of a larger platform migration effort. Output devices that were commonplace in the legacy environment (for example, high-volume Xerox or IBM mainframe printers) may not exist in the new environment. As a result, document formatting that was simple in the past may present a great challenge for modernization teams.

THE GOLDEN RULES OF SYSTEM MODERNIZATION

A system modernization team, like any group of IT professionals, proceeds in accordance with best practices and time-tested techniques. Yet several pieces of advice from outside the world of computing can help teams achieve successes beyond their immediate project objectives.

First, do no harm…

There are two main reasons why an organization would choose to modernize an existing application instead of replace it with a packaged solution. First and foremost, the current system likely provides unique functions or benefits that are not available in packaged systems. In addition, one advantage of an existing system is that the organization's employees, customers, and other stakeholders already know how to use it. The less a new system differs from the one it replaces, the shorter the learning curve for users and other stakeholders.

Just as a doctor would not attempt to heal a broken wrist by removing a patient's arm, modernization teams must ensure that a new system does not sacrifice important business functionality present in the existing system. Regarding document generation, management, and delivery, the new system should be able to produce and securely deliver all documents that are needed in critical business processes or which are required for regulatory compliance.

As previously discussed, one cannot expect a modernized system to perform every function in exactly the same manner as a legacy system. For example, an insurance agent may find that a customer quote letter formerly generated on a discontinued mainframe printer now arrives as a color PDF-formatted email attachment. However, the business need is the same, the information is the same, and the letter is the same. As long as the agent has the ability to print the letter on an alternate device or send it to their customer electronically, the core business requirement can still be fulfilled.

Learn from past lessons

In the end, there is one basic reason why system modernization is necessary: application code designed for one platform or operating system generally will not run natively in other environments. Said differently, if yesterday's applications had been written with portability in mind (following POSIX standards, for example), there would be less need to modernize them today.

But modernization projects are not just a necessary evil. In fact, they represent great opportunities to gradually increase the portability and sustainability of systems going forward. By moving away from proprietary or platform-specific technologies and embracing open standards, companies can dramatically improve both system longevity and flexibility.

When it comes to enterprise documents, there are a multitude of proprietary technology hurdles to overcome. For example, legacy applications may have been designed to produce output for a specific printer, fax server, or other output device. Very old systems may rely on special pre-printed stock in order to correctly print documents. Mainframe systems in particular were often designed to send print data over IBM's proprietary VTAM (SNA) protocol instead of today's more widely-accepted TCP/IP standard.

Any output management system deployed in the course of a system modernization effort should:

- Be able to run on a variety of hardware platforms and operating systems
- Provide modular conversion from nearly any incoming data stream to any target format for delivery to printers, electronic archives, email systems, etc.
- Support electronic document formatting, thus eliminating reliance on pre-printed forms
- Relieve business applications of the burden of directly managing printers and electronic document delivery

In short, modernization projects offer a rare chance to "undo" past decisions that often hampered an organization's ability to face new business and financial realities. Those who cannot learn from history are doomed to repeat it.

Seize the day

Modernization projects offer a chance to both correct past shortsightedness and to expand functionality for the future. Though major system enhancements are often outside the scope of an application modernization effort, planners should be prepared to exploit any and all "free" opportunities to improve their systems.

For example, organizations that include output management in their modernization plans often receive an unexpected windfall in the form of greatly increased delivery options. Unlike the purpose-built systems they replace, modern output management systems typically support email delivery, PDF document conversion, web document viewing, and other modern technologies. These options enable a modernization team to preserve and even extend the capabilities of proven legacy systems — with little or no additional effort.

LRS® OUTPUT SERVER COMPONENTS AND CAPABILITIES

To appreciate the role of an output management server in a system modernization effort, it is important to understand the basic functions and construction of the solution. The LRS® Enterprise Output Server solution is designed to provide all applications — whether modernized, packaged, or scratch-built — a robust central point of control for all system-generated output. The heart of the LRS output server is the VPSX® solution.

VPSX software provides a complete output management solution for enterprise applications. Its highly scalable architecture is designed to accommodate all POSIX-compliant environments ranging from single-department solutions to multi-server global systems. All elements of the product suite implement a single-process/multi-threaded design to ensure efficient use of system resources and enable VPSX servers to handle a very large number of input sources, output destinations, and end users.

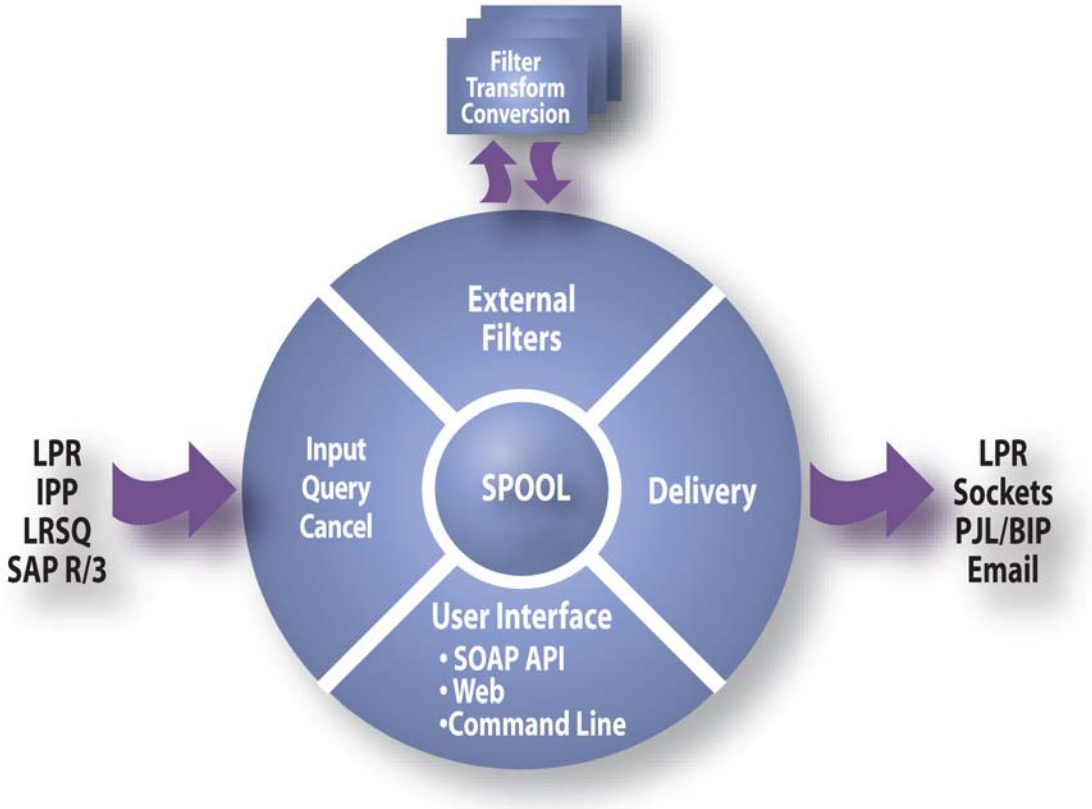

Figure 1 – Conceptual structure of the LRS Enterprise Output Server

A VPSX system is made up of several components. Users and administrators access VPSX functions via a web interface. The following diagram describes the functions of, and relationships between, each of the components:

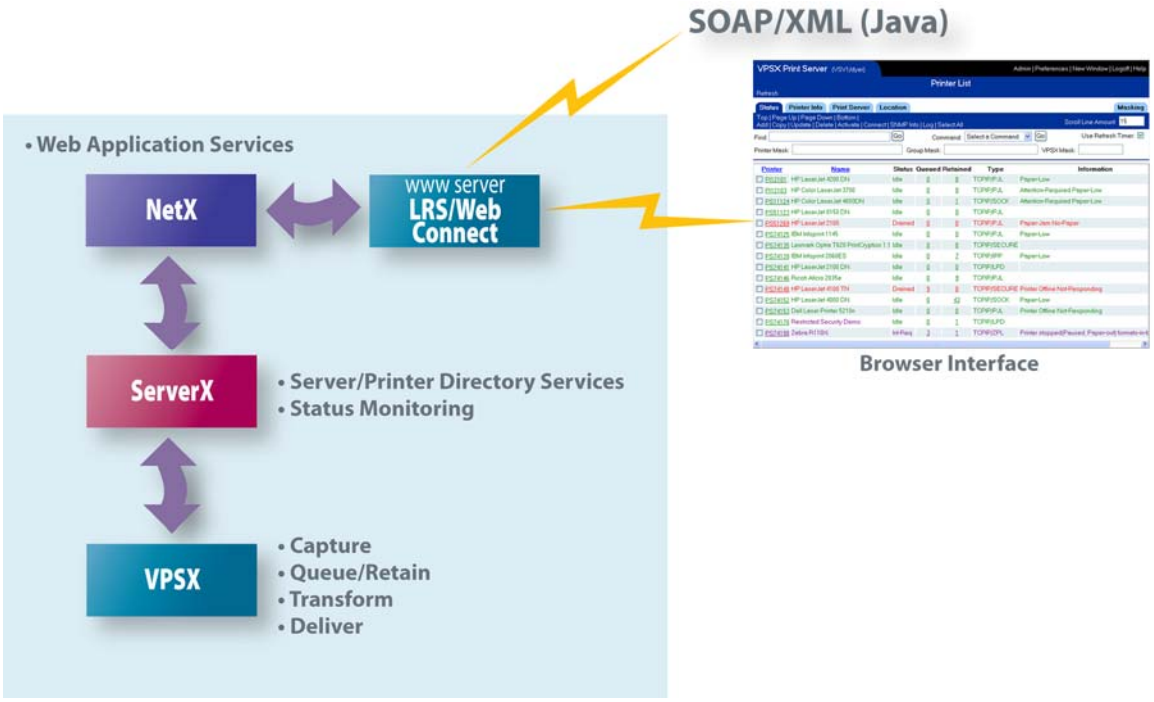

Figure 2 – VPSX Output Management Server Components

VPSX Print Server Component

An organization's printing environment can consist of one or more VPSX print servers. Since each print server is capable of managing a very large number of printers, the decision to run multiple servers is normally dictated by geographical location or operational units. Each print server is capable of independent operation but is managed by one or more LRS/ServerX servers. Each VPSX print server can register with one or more LRS/ServerX servers to enable central control and a single view of all printers across the enterprise.

LRS/ServerX Component

The LRS/ServerX component provides a central directory of all VPSX servers and their associated printers. All VPSX print servers that are registered with LRS/ServerX provide continual updates on the status of all devices. This enables LRS/ServerX to build a single view of all printers throughout the enterprise. LRS/ServerX then acts as the focal point for all access to VPSX servers and printers. Before accessing any resources, all users must authenticate with the LRS/ServerX process, which then controls access to all printers and servers based on the user's security profile. The LRS/ServerX process supports internal user authentication or can integrate with external security servers via the PAM (Pluggable Authentications Modules) interface to implement a single sign-on across all platforms.

LRS/NetX Component

The LRS/NetX component is a multi-threading Web application server that provides a common execution environment for the VPSX Web applications. Each of the standard components executes as a single process and attaches threads to process individual units of work. For this reason, VPSX software consumes few memory and CPU resources; a small server configuration can provide support for a relatively large number of printers.

VPSX software has been designed to enable print processing to be spread across multiple systems. SNMP traffic can be confined to remote locations and, at the same time, allow centrally located administrators to monitor and control all printers within the enterprise.

The following diagram shows a more complex configuration:

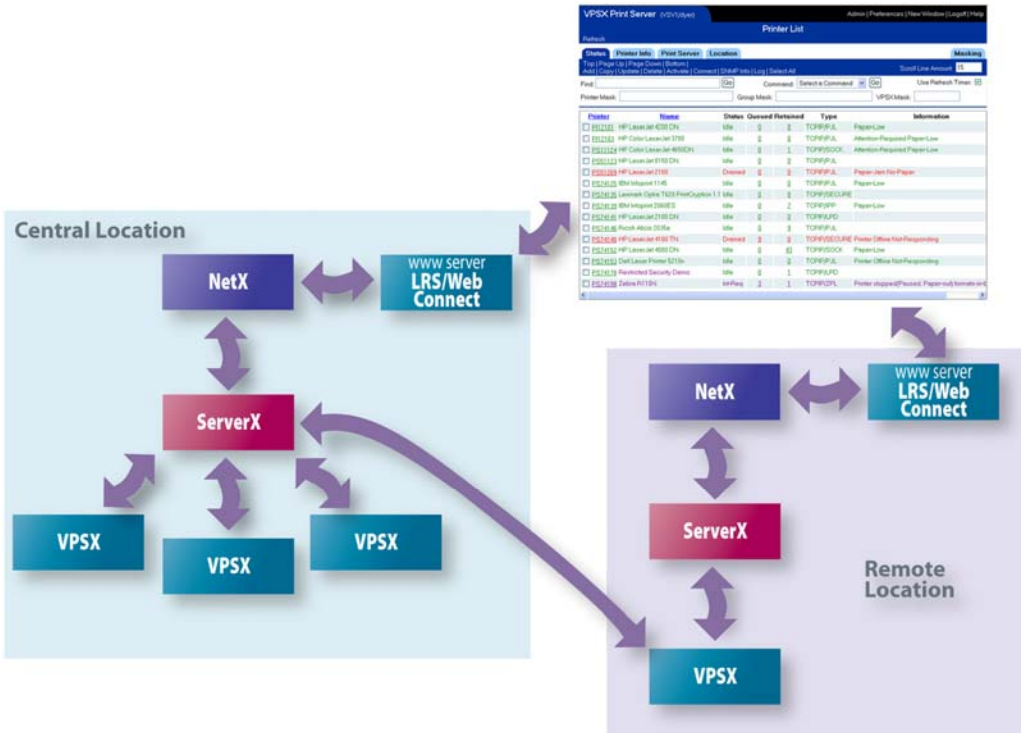

Figure 3 – Complex multi-server VPSX configuration

In this configuration, multiple VPSX systems are connecting to the centrally defined server. This enables all printers to be monitored via the central network interface while allowing remote users to control a sub-set of printers defined at the remote location. SNMP traffic between VPSX systems is limited to the locations where the VPSX systems are running.

The remote VPSX system will pass printer status information via the connection to the central server rather than having to define each printer to a central VPSX instance. This decreases the amount of SNMP traffic flowing across the network. Each VPSX system can operate autonomously and is not affected by the failure of other components. The VPSX systems running at the central location can be run on a variety of different operating systems or servers that operate different 'failover' technologies.

VPSX print data capture

Application modernization environments often involve numerous applications developed using different technologies. The ability to collect or capture data emitted by these applications is critical, and this is the essence of the VPSX integration with Micro Focus.

The VPSX solution provides numerous techniques for data collection or capture. Each technique provides for specific print job attribute information concerning the print request. Numerous third-party application vendors have imbedded an LRS agent in their applications to facilitate print job submission. Windows provides a simple to use interface that allows a VPSX system to 'take over' printing for a typical Windows user. These are the available interfaces:

- **LPR/LPD:** Supports LPR/LPD protocol. Capable of limited job attribute information via the LPR control file. TCP/IP based.
- **LRSQueue:** LRS transmission agent. Provides encrypted and compressed job transmission and a great deal of job attribute information. TCP/IP based.
- **JAVA J2EE IPP client:** LRS transmission agent. Provides job transmission and a great deal of job attribute information. HTTP (TCP/IP) based.
- **Windows IPP:** Microsoft job transmission agent. Capable of supplying some job attribute information. HTTP (TCP/IP) based.
- **Windows COM object:** LRS-developed COM object that can be used to create a print job from a Windows application written in practically any language.
- **LRS Port Monitor:** LRS job transmission agent. The port monitor makes use of standard Windows printing and the LRSQueue protocol. Capable of supplying some job attribute information.

These methods are fast and reliable. No matter which technique is used to collect or capture print jobs, VPSX software is able to dynamically determine print job attributes from the actual print data. In addition to the data stream format (PDF, PostScript, PCL5, PCL6, ZPL, AFP, JPEG, TIFF, etc), the software determines the page size, total number of pages, the presence of color and whether the document is printing duplex or simplex. These job attributes can be used to determine how the print request is processed.

The Micro Focus interface makes use of the LRSQueue transmission agent executing as part of either replacement code for redundant z/OS software or using the Micro Focus JES emulation.

VPSX print distribution

VPSX software can distribute print data via a wide method of protocols to remote printers, high-capacity print devices and servers. The transmission protocols for printing are:

- **LPR:** Standard protocol supported by many servers and printers in a modernized environment. Data is not compressed or encrypted, and there is no guarantee of print delivery.
- **Direct Socket:** Connection is established on a specific port number, and print data is sent directly to the output device. There is no guaranteed print completion.
- **BIP/PJL:** Exploits the Printer Job Language (PJL) capabilities of many modern devices. This enables enhanced error reporting and recovery, and provides guaranteed document delivery.

- **IPP:** Internet Print Protocol (IPP) enables print submission over HTTP to IPP-capable servers and devices. IPP provides enhanced feedback from output devices and better performance than BIP/PJL in some circumstances, but does not provide guaranteed document delivery.

In addition to delivering output to printers, the VPSX server also enables output to be sent via email, viewed via a web browser interface, or converted to PDF file form.

Administration and User Access

The VPSX interface for users and administrators is a Web (HTTP/HTTPS) based tool that requires no software to be installed at a given workstation. The interface works with any browser and is simple to use. The 'look and feel' of the interface can be modified to match an organization's 'corporate identity'. The tool makes use of the LRS/WEB Connect component and provides a consistent view of the VPSX print server no matter the underlying execution platform of any of the VPSX components. The views are broken into 'tabs' that provide further information when selected.

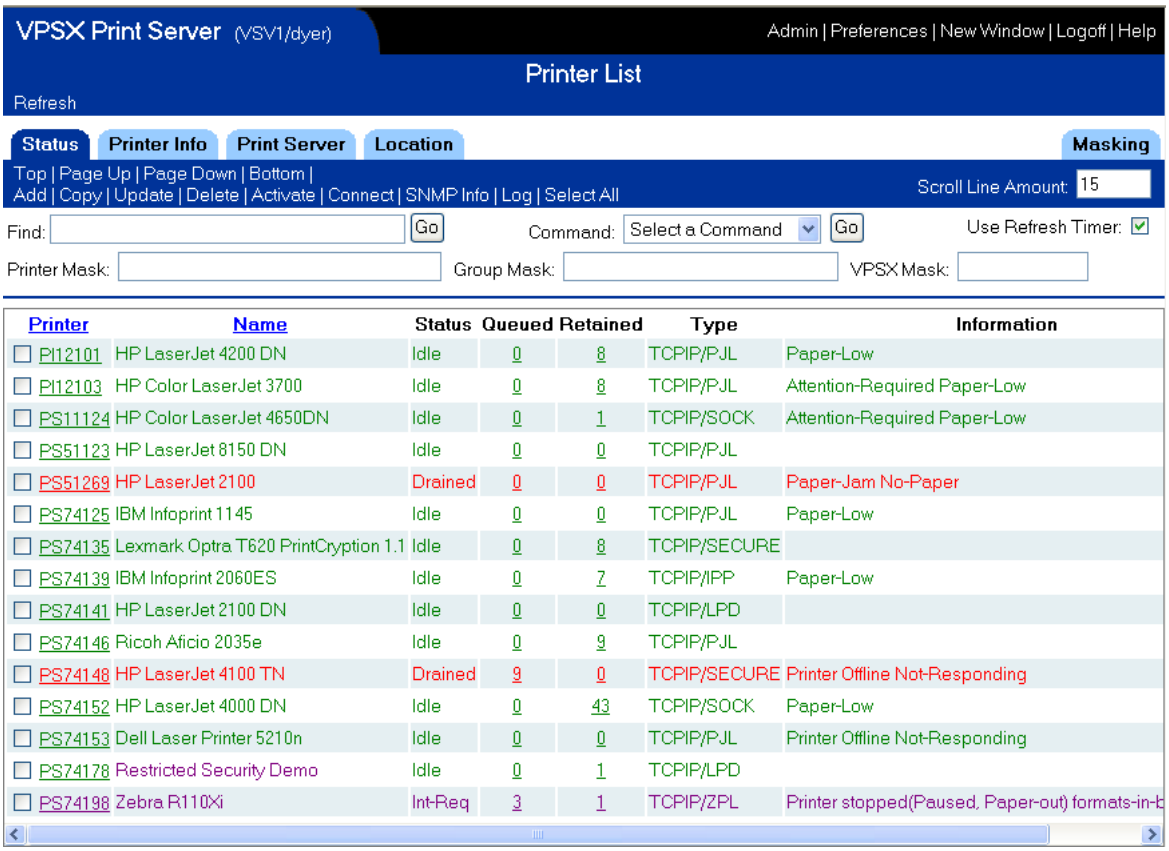

Figure 4 – The VPSX interface gives administrators web-based control over all output

Data Stream Conversions

As previously discussed, one major hurdle facing any modernization effort is the need to print or view system-generated output when legacy hardware no longer exists in the new

environment. To avoid this problem, the VPSX solution works with data stream conversion modules that transform legacy output formats to almost all popular modern formats.

Table 1 outlines some of the typical conversions used by the VPSX solution during the course of modernization projects. However, additional data stream conversions can be implemented when required to meet the needs of a specific project.

		Target Data Streams ("to")				
	Conversions	**PCL**	**PostScript**	**TIFF**	**PDF**	**AFP**
Legacy Data Streams ("from")	**AFP**	X	X	X	X	
	Mixed-mode data	X	X		X	
	LCDS (Xerox)	X	X		X	X
	PCL			X	X	X
	PDF	X	X		X	X
	Image File Format	X	X		X	X

Table 1 – Select data stream conversions used during system modernization projects

VPSX data stream conversion support eliminates the need to re-write legacy application code, even when the original devices are unsupported on the modernized platform. For example, mainframe applications may have been designed to generate LCDS, Metacode, or AFP formatted documents and transmit them to a high-volume printer via a Bus/Tag interface. The VPSX solution enables this same application code to send these documents to standard TCP/IP devices. This ability to seamlessly convert data streams helps VPSX software preserve a company's investments in proven legacy applications and the documents they create.

The Sum of the Parts

Each of the individual VPSX components is designed to provide maximum performance and flexibility. Together, they form a scalable central point of control that supports a wide variety of network protocols, data streams, and hardware platforms. Since modernization projects aim to bridge the gap between legacy technologies and emerging ones, the importance of this flexibility cannot be overstated.

The open standards-based approach of the VPSX solution protects customers' current IT investments from becoming obsolete as future technologies emerge — thus reducing and delaying the need to modernize the same system in the future. To help customers achieve their modernization goals, LRS has developed a series of custom interfaces that facilitate communication between the Micro Focus Net Express server and VPSX solutions. These interfaces are described in greater detail in the following section.

INTEGRATING LRS AND MICRO FOCUS SOLUTIONS: INTERFACES AND TECHNICAL DETAILS

The VPSX solution helps IT personnel efficiently manage printing in environments that have migrated applications to run on a Net Express server. Application programming interfaces supplied by independent software vendors (ISVs) enable legacy applications to execute on Net Express server platforms. Integration between VPSX software and the Net Express environment involves the following components:

- Legacy COBOL applications
  (Requires no change to either the application or the MVS JCL)
- Legacy ISV print programming interfaces
- Micro Focus Net Express JES emulation
- LRSQueue protocol
  (Print job submission to the VPSX solution from within Micro Focus Net Express)
- VPSX software from LRS

The LRS solution can integrate with seven different types of migrated legacy programs:

- CICS/COBOL programs using the CA-SPOOL ESF interface
- CICS/COBOL programs using the LRS DRS/API
- BATCH COBOL programs using the CA-SPOOL ESF interface
- BATCH COBOL programs using the LRS DRS/API
- BATCH COBOL programs using SYSOUT DD statements
- BATCH COBOL programs using the CA-SPOOL SUBSYS DD statement
- CICS/COBOL standard EXEC CICS SPOOLOPEN interface

Application programs require no changes; they can simply be re-built without modification. The various methods of integration use either ISV API-specific control blocks or Micro Focus-supplied control blocks to pass the necessary information, enabling VPSX software to effectively process the spool file. The information consists of fields passed in the relevant control blocks such as:

- JOBNAME
- OWNER (USERID)
- DESTINATION
- FORM
- FILENAME

These fields are used to dynamically build LRSQueue commands that are then executed using a built-in Micro Focus COBOL function that allows for the execution of any Windows or UNIX based command.

CA-SPOOL CICS/COBOL and BATCH COBOL interfaces

The CA-SPOOL ESF API interface uses multiple entry points or modules to enable the creation of a spool file. The supported entry points are:

- **OPEN** – initiates the creation of a spool file in CA-SPOOL
- **WRIT** – writes a spool file record
- **CLOSE** – completes the closing of the spool file in CA-SPOOL, thus making it available for processing by CA-SPOOL

In the z/OS environment, control information is normally passed between each request. The OPEN request causes CA-SPOOL to return control information used for subsequent requests related to this spool file. This information is passed back via the OPEN-CBA field. It is the responsibility of the application program to pass this control information in the parameter lists associated with the WRIT and CLOSE requests.

The VPSX integration components mimic the mechanism used by CA-SPOOL in that the OPEN-CBA field passed by the application program is used to store the address of control information between each print request. The storage is acquired in the OPEN request and returned via the OPEN-CBA field.

As a result, application programs that use either the CA-SPOOL CICS/COBOL or BATCH COBOL interfaces can be made to interoperate with VPSX software ***without having to alter or manipulate the original COBOL source code***. This dramatically simplifies the task of migrating application programs that create printed output. At the same time, the migrated system can still use the printer names known to the original applications, since the VPSX solution supports practically an unlimited number of printers.

LRS DRS/API CICS/COBOL / BATCH COBOL interfaces

The LRS DRSAPI interface makes use of multiple entry points or modules to enable the creation of a spool file. The supported entry points are:

- **INIT** – initiates a spool file in JES2 or JES3
- **PUT** – writes a spool file record
- **TERM** – completes the closing of the spool file in JES, thus making it available for processing by the mainframe-based VPS output management solution

The principle difference in this mechanism is that LRS make use of the JES spool. The DRS/API makes use of dynamic allocation (DYNALLOC) to create spool files in JES, based on fields passed via the DRS DRRB and DRIB control blocks to the SVC99 parameter list that is used to create the file.

Again, in the z/OS environment, control information is passed between each request. The INIT request causes DRS/API to return control information that must be used for subsequent requests related to this spool file. This information is passed back via the DRS-DRRB-REPORT-ID field in the DRRB. It is the responsibility of the application program to pass this control information in the parameter lists associated with the PUT and TERM requests.

The VPSX integration components mimic the mechanism used by CA-SPOOL in that the DRS-DRRB-REPORT-ID field passed by the application program is used to store the address of control information between each print request. The storage is acquired in the INIT request and returned via the DRS-DRRB-REPORT-ID field.

As a result, application programs that use either the DRS/API CICS/COBOL or BATCH COBOL interfaces can be made to interoperate with the VPSX solution without having to alter or manipulate the original COBOL source code. Again, this makes it easy to migrate application programs that create printed output. At the same time, the migrated system can use the printer names known to the original applications, as the VPSX system supports practically an unlimited number of printers.

BATCH COBOL programs using SYSOUT DD

Most spooling mechanisms in z/OS systems rely upon the JES spool in order to create output. The same is true of the Micro Focus Net Express environment in that Net Express can emulate JES, thus enabling administrators to run exactly the same JCL without change.

JES normally uses the DESTINATION field on a SYSOUT DD statement to prompt the distribution of a SYSOUT dataset. In JES, there are a nearly unlimited number of DESTINATIONS that can be defined, ranging from R numbers, U numbers and esoteric DESTINATIONS. This enables products such as the VPS solution to utilize DESTINATION values to deliver output to many thousands of printers.

By contrast, the Micro Focus implementation of JES spooling allows for the assignment of a MSGCLASS and the association of this MSGCLASS with a Windows-defined device. Thus, the native Micro Focus implementation can only print to a very limited number of devices, because a MSGCLASS is associated with a specific printer when the printer is ‘defined’ to the NET Express server.

The VPSX interface removes this limitation by using the DESTINATION value rather than the MSGCLASS to determine where a document should be printed. This is achieved by associating one or more printers with one or more MSGCLASSES and a Net Express printer exit. The printer exit simply sends the print job to the VPSX solution, which uses the DESTINATION value to determine the name of the print queue to send the job.

BATCH COBOL programs using CA-SPOOL SUBSYS DD

CA-SPOOL provides an alternate mechanism to allocate spool files. This mechanism relies upon the z/OS CA-SPOOL started task being active – a situation that is impossible to duplicate in the Net Express environment. Micro Focus provides a facility whereby any spool file allocated using a CA-SPOOL SUBSYS DD statement will be passed to the standard printer exit. The SUBSYS DD statement is addressable by the exit and is used to determine the name of the destination print queue.

Underlying technology

The interfaces make use of several built-in functions available to COBOL programs running on the Net Express platform. These functions are externalized in the modified routines called by the original COBOL application programs. The replacement routines greatly reduce the complexity of the application, as they are developed in COBOL.

CONCLUSION

In today's business environment, IT managers are often asked to choose between their time-tested existing systems and the promise of newer, more cost-effective technology platforms. By modernizing their existing applications — and the documents they create — companies are able to have the best of both worlds.

Application modernization projects give organizations the opportunity to dramatically improve both the quality and cost-effectiveness of their document systems. LRS has developed customized interfaces to help Micro Focus customers more quickly migrate their legacy system output to the VPSX output management solution. This standards-based solution provides a scalable, platform-independent central point of control for all of an organization's business critical output.

As with any IT project, a system modernization project represents a significant investment in time and effort. Micro Focus and LRS offer the tools and expertise to help companies more quickly realize the benefits of system modernization. To learn more about how we can help your organization, contact LRS at: microfocusalliance@LRS.com.