# Xerox® FreeFlow® VI eCompose Software

User Guide

**xerox**™

# Contents

Contents

# FreeFlow VI eCompose Installation

This chapter contains:

🖉 Note: Refer to the following information:

- Install locally

  Install locally VIeC and all components. You are required to log in locally, with system administrator credentials. Installations from a remote location, or using Windows Terminal Services, are not supported and do not work as expected.

- Virtual machine

  You can install VI eCompose on a virtual machine with a unique MAC address or UUID. However, no support is provided for the VM environment.

- Updates are released as Full Installers.

- You can install any FreeFlow VI Suite 18.0.0 update over any prior release of the FreeFlow VI Suite. Installation of VI Suite components require system administrator privileges.

- Before any update installation, stop all active VI eCompose modules.

- Before any update installation, stop all active VI eCompose modules, such as FreeFlow VI eCompose or VI eCompose services.

## Getting Started

Before you begin, first download the software from the Xerox website. Download instructions are provided here. After you have downloaded the appropriate files, you can proceed to the installation instructions that follow.

### PROGRAM DOWNLOAD

To download VI eCompose software, go to www.xerox.com/support, then select **Software & Platforms**. Select **FreeFlow > FreeFlow Variable Information Suite**, then select **Software and Solutions**. If necessary, to display the correct installer file, select the operating system appropriate for your target platform, then download the VI eCompose installer file.

> Note: Programs are delivered in .zip format. You can download the programs directly to the target device. If the device does not have Internet access, copy the .zip file on the target device. Extract the zip file, which contains .exe file for the FreeFlow VI eCompose installation.

### DOCUMENTATION DOWNLOAD

English language versions of the FreeFlow Variable Information Suite documentation is available for download. Go to www.xerox.com/support, select **Software & Platforms**, select **FreeFlow**, select **FreeFlow Variable Information Suite**, and select **Documentation**.

### BARCODE FONT DOWNLOAD

To download barcode fonts, go to www.xerox.com/support, select **Software & Platforms > FreeFlow > FreeFlow Variable Information Suite > Software and Solutions**.

> Note: When downloading fonts, you are directed to review an End User License Agreement. To download the fonts, review and accept the End User License Agreement. If you do not accept the End User License Agreement, you exit from the font download page.

### SPECIALTY IMAGING

Xerox Specialty Imaging is not supported when generating PDF as the output format. VIeC does not support any Specialty Imaging effects.

## Important Upgrade Information

If you are using a version earlier than VI eCompose 16.0.3.0 and you are upgrading to a new version, before you uninstall the earlier version, install version 16.0.3.0 or later.

To enhance security, the internal password hash algorithm for the eCompose Web Application (HTTP) and Secure Web Application (HTTPS) have been upgraded from 256-bit SHA-2 to PBKDF2 with HMAC-SHA256. As a result, user passwords generated on versions of eCompose prior to version 15.0 no longer function, and before users can log in again, a one-time password reset is required.

A new password utility has been provided to help in this transition. You can start the utility from the command line or from a batch file and works by resetting the indicated user password in the Web application password file (vtpw. pwd). It is recommended that the administrator uses this utility to reset each user password to a unique temporary password. When the password is reset and provided to the user, the user logs in using the temporary password and immediately changes the password to a password of their choice.

### RESETTING A PASSWORD IN VIECOMPOSE VERSION 18.0.X

To reset a password in VIeCompose version 18.0.x, perform the following steps:

1. Open the Command Prompt.

2. Change the directory to `X:\Program Files (x86)\Xerox\VIPP\xvtp\bin`, where X is the disk partition in which the Xerox® FreeFlow® VI eCompose software is installed.

3. Enter the following syntax to change the password of a user: **Changepassword. exe<username><password><port>**, then click **Enter.**
   The following are the definitions:

   - username: Name of the user

   - password: The new password for the user

   - port: The port number, if the default port is changed.

   Example:

   ChangePassword.exe test test@123

   ChangePassword.exe test test@123 9000

   .

## vipo.ini Configuration File Information

The configuration of the VIeC installation is specified in the `vipo.ini` file at `C:\ProgramData\Xerox\VIPP\VIPO\vipo.ini` for Windows®10, Windows®11, Windows Server® 2016, Windows Server® 2019, and Windows Server® 2022.

The vipo.ini file is similar to this example:

```
[_]
relver=14.0
server_root=C:\xvtp
user_root=C:\xvtp\users
remote_server_root=N/A\xvtp
remote_user_root=N/A\xvtp\users
language=en
log_options=4
xgfdir=C:\xvtp\bin\xgf
xgfcdir=C:\xgfc
media=USLetter
distiller_root=c:\xvtp\bin\normalizer
security_options=C:\xvtp\securityoptions\Default.SecurityOptions
enable_pdf_security=false
use_ps_dsc_conform_tag=1
shorten_job_folder_names=4
enable_job_folder_names_with_msec=1
enable_distillation_trace_log=2
vtp_distill_proc_wait_timeout_hours=24
vtp_distill_page_wait_timeout_minutes=60
split_proc_wait_timeout_hours=24
split_count_wait_timeout_minutes=60
output_filename_strict_checking=true
distill_and_split_without_bookmarks=error
split_invalid_bookmark_action=1
split_transfer_attributes=true
display_split_warnings=false
jsc_cancel_job_scope=2
csv_newline_char_seq=(&#x0A;)
csv_dquotes_char_seq=(&quot;)
vtpd_log_options=4
vtpd_incoming_folders_filter=IncomingFolders.filter
vtpd_auto_run_filter=AutoRun.filter
vtpd_pending_jobs_lo_water=5
vtpd_pending_jobs_hi_water=20
vtpd_max_jobs_in_completed_queue=1000
vtpdispatch_max_preemption_count=0
v2vwatch_poll_interval_seconds=5
v2vwatch_max_timeout_check_minutes=10
v2vwatch_file_size_check_period_seconds=60
v2vwatch_file_readiness_check_method=check_file_size
normalizer_options_file=<default>
split_match_passwords=warning
split_short_individual_open_password=warning
enable_multi_instance=1
vtpserver_number_instances=1
vtpdispatch_number_instances=1
lsa_type=4
job_selection_method=fifo
server_name=7JD04X1_0
```

A key entry is `user_root=`

- A VIeC Server processes any job submitted with user_root.

- A VIeC Job Submission Client submits local jobs to the appropriate location with user_root.

- The VIeC installation program adjusts the user_root entry according to the configuration that you select:

    – VIeC Server and Client

    – Client only

    – VIeC Server Node servicing a VIeC Cluster

    🖉 Note: The use of `remote_user_root` and the associated `remote_server_root` entries were intended for use with a developer license configuration that is no longer available. Both entries are deprecated, so ignore the entries mentioned in older or other external documentation.

    🖉 Note: Manual edits of the `.ini` and `.conf` files are discussed in the related sections of this guide.

## System Requirements

VIeC can be installed on workstations that run in these environments:

> 🖉 Note: For processor, memory and disk requirements, refer to the System Requirements for installing the Windows operating system.

- Windows® 10, Windows® 11, Windows Server® 2016, Windows Server® 2019, Windows Server® 2022, and 64–bit platform only

- A screen resolution of 1024 x 768. The higher resolution is preferred.

- DVD-ROM or CD-ROM

- Microsoft .NET Framework 4.8 (installed)

**Requirements to Run Multiple Instances of VIeC**

For best results, a minimum of one CPU core and 2 Gbytes of RAM is required for each instance of VIeC. If Dispatch is involved, enough CPU or memory for all simultaneous post-processing operations is required. The requirements vary, based on the load and complexity of the jobs that need processing.

A minimum configuration for four instances, without considering VIeC Dispatch, is a quad-core system with at least 8 GB of RAM, running on a 64-bit operating system. If there is Dispatch post-processing or other system load, more CPU and RAM capacity is required.

**Requirements to Run the VIeC Web Job Submission Service**

The general rule is that each Normalizer instance determined by the `vinip.ini` setting nip_number_of_ instances requires one CPU core and 2 Gbytes of RAM, just as with the initial VIeC multi-instance configuration. After the CPU and RAM allocations are made, further adjustments can be required. A dual-core system with 4 GB of RAM would initially have nip_number_of_instances set to 2.

The expected use of WJSS is the submission of many small jobs with a few pages. WJSS can support the submission or any job, small or large. This can mean that the memory footprint requirement for a given Normalizer instance is less than 2 GB. A high volume of small jobs can put a different load on network and disk I/O, which can push the overall system to being more I/O bound than CPU bound. When the system is up and running, users can see benefits from running more instances than there are cores, or they may need less working RAM than a system with different workflow requirements. On-site experimentation is encouraged.

**PDF/VT-1**

The Additional Profile Options dialog box available from the Job Submission Client includes an option to create PDF/VT-1 compliant PDF files that include media and finishing options. When using PDF/VT-1 and media and finishing options, always consider this information:

- Media and Finishing options and PDF/VT-1 compliance are applied only to the Master PDF file, and not to any child PDF files that are split from the master PDF. While the master PDF file can include Media and Finishing options and be PDF/VT-1 compliant, any child PDFs split from the master PDF do not have such features included.

- Not all printer devices honor PDF files that contain Media and Finishing requests or that are compliant with the PDF/VT-1 standard. Only the digital front end (DFE) systems that are supported honor the feeding and finishing requests.

### Supported Print Servers

Xerox FreeFlow Print Server (FFPS) version 9.3 SP1 or above.

### Trial Period

The initial installation of VIeC provides a 60-day trial period. After the trial-period, a production license is required for system functionality. You can activate the production license anytime during or after the trial period.

> Note: The trial detects prior installations of VIeC and cannot activate the trial if the software was installed more than 60 days before. Do not change the system clock to circumvent the license mechanism because VIeC detects changes and cannot operate on that system.

The System Host ID of the Windows workstation and the product release version are required to get a license. When a license is not available immediately, the software can be installed, and the license can be activated later.

System administrator login credentials and privileges are required to install the VIeC software. Modify/Read and Execute/Write privileges are required on the drive and directories where VIeC is installed.

## VIeC Installation

1. Download the VI eCompose .zip file, then extract the files to get the .exe file for the installation. For more information on downloading VI eCompose, refer to Program Download.

2. Run the .exe file. To install the software, follow the onscreen instructions.

   a. Accept the Software License Agreement.

   b. Choose the following options from the install set:

      - **Client and Server**: This option requires a software license.

      - **Client Only**: This option requires access to a remote server.

      - **Server Node**: This option services a cluster.

      - **LPR Setup**: This option appends the LPR feature to an existing VI eCompose installation.

      - **VIeC Web Application**: If you want to submit jobs through Web, select the option. IIS will be enabled as a part of FreeFlow VI eCompose web application setup.

      - **LPR Job Submission**: If you want to submit jobs through LPR, for LPR print driver installation, select **Yes**.

3. When prompted, provide the following VIeC configuration information for your system:

   - Choose the Default Job Folder name.

   - Choose the drive location for the program and customer files.

   - Choose the drive that contains VIPP® customer files, such as the `xgfc` directory.

   - Choose the available VIeC Servers on the network.

   - Choose the default media size: A4 or US Letter.

   - Choose the shortcut folder to create a product icon.

   For more information about the available configuration options, refer to VIeC configuration options.

4. After the installation completes, restart your system.

To convert the trial license to a product unit, order the VI eCompose License kit. After the order is processed, the license kit is delivered. Included in the license kit is the software activation key. To activate your software, follow the instructions in Using the Activation Key Option. Not all locations across the world support the use of license enablement using the automated license activation process. If the license kit does not contain the software activation key, obtain a license file, then follow the instructions in Using the Load License Option.

## VIeC Licensing

If the software has not been installed previously on your device, VI eCompose software installs with a 60-day trial period. To use the product after the 60-day trial period, to convert the trial to a fully licensed version of the product, purchase a production license. Without a valid license, the software is not usable.

The process to license VI eCompose depends on where and how VI eCompose was purchased. VI eCompose software has unique licensing requirements based on the geographical region. When you purchase VI eCompose software, the licensing requirements for your region are included in the kit.

- System clock

  Do not attempt to change the system clock to circumvent the license. VIeC software detects changes. If tampering is attempted, the software fails to operate.

- Manual Load License process

  When using the manual process of the Load License option, you are provided a license file. License files have the file extension *.dat for all VI products that are generated using an automated process. The process saves the file as `HardwareAddress.dat`, where HardwareAddress is the address of the computer on which the product is licensed. It is possible to receive more than one product license file with identical file names. It is your responsibility to ensure that you do not overwrite existing licenses when saving a new product license. When license files that are used on a single computer are set to expire at the same time, you can request one file that activates all VI products on that computer.

- VIeC Clients do not require a license.

- A workstation in a client-only configuration accesses a remote, licensed VIeC Server to process VIPP® jobs.

  Note: When you install an upgrade to an existing version of VIeC software with a valid license, the software finds and activates the previously installed license. No further action is required until the license expires.

To manually load or activate your VIeC license, from the menu bar on one of the main Dispatch, Local, or VIeC Server panels, select **Help > Install/Update Software License**. You can use this option to view how many days remain in the trial version or view the status of the license.

When you select the Install/Update Software License option, a VI eCompose License panel appears. The panel provides access to information about the installed license and two unique licensing options. Use the panel to choose the option suited to your location. The panel includes license information:

- License Type

- Days Remained: The number of days until the existing license expires.

- License Host ID: VleC software finds and displays the information from your computer. If a valid license file has been loaded, the Host ID used by the license is shown.

- Product Version: The installed version of the VIeC software.

If you have obtained a license file: Use this option when you have received a license file with the file extension *. dat. When the file is delivered, ensure that you store the file in a safe and accessible place on your computer.

For instructions, refer to Using the Load License option.

If you have an activation key: The information in the panel is designed for users who have received a software activation key as part of the software license kit for the VIeC software.

If you do not have the license kit, and are in a location that requires this option, to purchase the software license kit

for the VIeC software, contact your Xerox sales representative. After the order has been processed, a license kit is shipped to you. The license kit contains the software activation key and serial number, if provided.

For further instruction, refer to Using the Activation Key option.

## USING THE LOAD LICENSE OPTION

To activate the VI eCompose license using a license file:

1. Select the **Load License** option.

2. Browse to the location of the license file, select it, then click **OK**.

3. The license will be installed. The new license information will be displayed at the top of the license panel.

4. Select **OK** to activate the license.

## USING THE ACTIVATION KEY OPTION

Note:

- Retain the software activation key and serial number in a safe location for future upgrades or support.

- If you have the software activation key, you can use the automated license process.

1. For System Host IDs, choose one of the available options for the license Host ID. The default is recommended.

Note: If the automated license process fails, when you contact Xerox Technical Support for assistance, have available the hardware System Host ID string and the software activation key.

2. For Serial Number, enter the serial number of the device where the software is installed.

3. For Software Activation Key, enter the information, then click **Activate**.

4. The Xerox License Server can request additional information. Enter the requested information, then select **OK**.

   This information and the Software Activation Key are submitted to the Xerox License Server.

5. If additional information is required to validate the license request, a new screen appears. Enter the required information, then select **OK**.

   The Xerox License Server validates the information and enables the license or returns an error code if the information supplied does not match the Xerox records.

6. If an error code is generated, for assistance, contact your local Xerox representative and report the error code.

7. If the information entered is valid, a license file is generated and loaded to your system. Details of the license appear in the License Information area of the License screen. To apply the license, click **OK**. When the OK button is selected, the license is applied.

Sometimes, the automated license activation fails. If the license activation fails, for assistance, contact Xerox Technical Support. Ensure that you have available the System Host ID of the workstation and the software activation key supplied in the Software Activation Kit. If the license is emailed to you, you can activate the license manually.

Note:

- If connecting to the license server fails, you can go to any Internet-connected device and log in to the portal using this URL: https://www.xeroxlicensing.xerox.com/activation.

- Enter the activation key, then follow the onscreen prompts. Ensure that you have available the System Host ID of the VIeC system. Xerox emails a license file to you. Move the license to a safe location on the VIeC system, then use the Load License option to install the license.

## VIeC Configuration Options

During installation, you are required to select the VIeC software configuration for your site. To determine which of the available VIeC configuration options is appropriate, click the links and review the information.

- VIeC Server and Client
- VIeC Client only
- VIeC Server Node servicing a VIeC Cluster

### VIEC SERVER AND CLIENT

This is the default configuration that results from a typical single-workstation VIeC installation.

Choose this option to install the following components on a single workstation:

- Local VIeC Server
- Job Submission Client
- Watched Folders Client
- VIeC Web Application
- VIeC Secure Web Application
- VIeC Dispatch Server

This option is discussed in detail in VIeC configuration options, a section of the *FreeFlow VI eCompose User Guide*.

### VIEC CLIENT ONLY

Choose this option to install the following components in a client-only configuration:

- Job Submission Client
- Watched Folders Client

No license is required. An existing VIeC Server is required on the network where the client can submit jobs.

> Note: The installation script requires that you select the Remote Server Drive. Before the path can be entered on the client installation, an available VIeC Server on the network is required. The client requires access to the server.

For more information about this option, refer to VIeC configuration options, in the *FreeFlow VI eCompose User Guide*.

### VIEC SERVER NODE SERVICING A VIEC CLUSTER

Choose this option to install a VIeC Server Node servicing a VIeC Cluster. This is an advanced option to be used by system administrators only. Refer to VIeC configuration options, a section of the *FreeFlow VI eCompose User Guide*, for more complete information.

## Additional VIeC Functions

Additional functionality can be added either during or after the initial installation of VIeC. To add the additional functionality, you must log into an account with administrator access on the PC. Insert the CD you created previously into the PC's CD/DVD drive. The installation should start automatically. If the installation does not start automatically, browse the CD to the Windows folder and double-click on the XeroxFreeFlowVIeCompose-18.n.n.exe file to start the installation. Follow the on-screen instructions to add additional functionality.

Installation of these additional functions are described in these sections:

- lpr job submission installed on a VIeC Server
- Additional VIeC printer creation
- VIeC Server Node installation
- VIeC Services installation

## lpr Job Submission Installed on a VIeC Server

To install the lpr job submission option for VIeC, select **lpr printing** during VIeC installation, or select **Install VI eCompose LPR Option** from the VIeC program folder after VIeC installation.

This option must be installed on each VIeC Client machine that will be running lpd and receiving lpr jobs for submission to VIeC for processing.

Note: To successfully install lpr job submission:

- Log on locally to the system where the VIeC LPR Printing installer will execute using a login account that has Administrator rights. Installations from a remote location, or using Windows Terminal Services, will not work as expected.

- Ensure that VIeC Client is installed on the Windows system that is receiving the lpr jobs.

- Ensure that you have Control Panel access.

Access to the Control Panel varies between Windows or Windows Server Operating Systems in use. Adjust these instructions according to your operating system.

1. From the Start button, go to **Settings > Control Panel**.

2. At the Control Panel, find and double-click **Administrative Tools**.

3. In the Administrative Tools window, find and double-click **Services**.

4. In the Services window, find **TCP/IP Print Server**. When the option is present skip to Step 8, when not continue to the next step.

5. Install TCP/IP Print Server. Go to **Settings > Control Panel > Add Remove Programs**.

6. Click on the **Add/Remove Windows Components** icon on the left.

7. Check **Other Network File and Print Services**, then click **Next** and **Finish**.

8. Return to **Control Panel > Administrative Tools > Services** and double-click **TCP/IP Print Server**.

9. Select **Automatic** for the Startup Type.

10. Click **Start**, then click **OK**.

## Additional VIeC Printer Creation

You can create additional VIeC printers.

✎ Note:

- Characters in printer name:
  The VIeC printer name cannot exceed 64 characters and must consist of only alphanumeric, dash (-), and underscore (_) characters. All other characters result in errors or unexpected outcomes.

- At the end of each step in the wizard, click the Next button as required.

- Access varies.

Access to programs and settings varies according to the Windows or Windows Server operating system in use. Adjust these instructions according to your operating system.

1. From the Start menu, go to **Settings > Printers**.

2. To launch the Add Printer wizard, double-click **Add Printer**.

3. Select **Local Printer**.

   ✎ Note: Ensure that this setting is not selected: Automatically detect and install my Plug and Play printer.

4. Select **xvtp port**.

5. For the manufacturer, select **Generic**. For the printer type, select **Generic/Text Only**.

6. When prompted, select **Keep existing driver**.

7. For printer name, enter the virtual printer name for the lpr-to-VIeC command.

   ✎ Note: Ensure that the virtual printer is not selected as the default printer.

8. When prompted, select **Do Not Share This Printer**.

9. To bypass printing a test page, select **No**, then click **Finish**.

## VIeC Web Application

Note: Access to programs and settings varies according to the operating system in use: Windows® 10, Windows® 11, Windows Server® 2016, Windows Server® 2019, and Windows Server® 2022. Adjust these instructions according to the operating system.

Note: FreeFlow VI eCompose version 18.0.x uses Windows Internet Information Services (IIS) to enable using a Web browser optionally to submit VIPP® jobs and perform administrative functions on the server. During installation, you are asked if you want to enable this feature. To start and stop the Web application, use Windows Internet Information Services manager.

To enable the VIeC Server to accept VIPP® jobs using the Web:

To submit VIPP® jobs to VIeC Web Application:

1. Obtain the system Network Identification or the IP address.

   - To obtain the Network Identification, select **Start > Settings > Control Panel > System**, then click the **Computer Name** tab. The system identification is the full computer name, for example: lab computer.

   - To obtain the IP address, from a command prompt, enter `ipconfig`. The IP address is the multiple-part number, for example: 11.1.1.111.

2. To access the Xerox® FreeFlow® VI eCompose browser interface, enter the URL address in the browser as `http://<Server IP Address>FreeFlowVIeCompose` or `https://<Server IP Address>FreeFlowVIeCompose`.

3. Enter the site and Click the **Submit Job** button in Job Status and Submission page. You can submit VIPP® jobs to VIeC Web Application.

> **Caution:** It is recommended that you do not leave the VIeC Web installation at the default level of security. For more information, refer to Defining VIeC Web security levels.

### DEFINING VIEC WEB SECURITY LEVELS

There are four types of security supported by VIeC Web:

| | |
|---|---|
| No security, HTTP only | Anyone with TCP/IP access to the VIeC Web Application can submit, view, retrieve all jobs, and perform VIeC Web and VIeC Server administration. The HTTP setting can be an acceptable level of security for a VIeC installation on a LAN that is not connected to a larger network, such as in a training environment. The No Security HTTP only setting is not recommended for a production environment. |
| HTTP Basic Authentication | Allows user account-level security with user passwords and group security on an area-by-area basis. The HTTP Basic Authentication option is preferred to the No security, HTTP only option. |
| | Note: Passwords transferred between the client and Web browser are transmitted as plain text and are not encrypted. |

| | |
|---|---|
| HTTPS (HTTP Secure) over SSL (Secure Sockets Layer) | The HTTPS option encrypts the traffic between client Web browsers and the VIeC Web Application. |
| | Note: Using SSL alone does not provide any user name, password, or group configuration capability. |
| HTTPS via SSL with HTTP Basic Authentication | This is the preferred level of security for a VIeC Web installation. HTTPS and SSL combine the user name, password, and group configuration capabilities of HTTP Basic Authentication with the transaction encryption provided with SSL. This security level is obtained by applying at the same time the Basic Authentication and SSL configurations. |

**No Security, HTTP Only**

This is the default security level at installation. Xerox recommends that you change this security level when the VIeC Servers are accessible from the internet and to protect sensitive information found in production environments.

**HTTP Basic Authentication**

The VIeC Web implementation of HTTP Basic Authentication allows the VIeC administrator to set up a system of user-by-user access to controlled areas or realms. The VIeC administrator can associate user names with access to a specific realm or with named groups that are designated as having access to a specific realm.

Security for non-web-enabled VIeC installations can be restricted by setting read and write access on a user or group basis. To set the read and write access, on the operating system where the VIeC server is hosted, use the permissions mechanism of the file system.

After a VIeC Server is web-enabled, the read and write access at the operating system file system level is performed by the VIeC Web Application. The VIeC administrator can provide equivalent security. The VIeC Web support for Basic Authentication is an important part of providing that level of security.

HTTP Basic Authentication can be used with HTTP using the vtpweb.exe application or HTTPS.

To enable Basic Authentication for VIeC Web:

1. Click **Basic Authentication Quick Setup**.

2. In the required fields, type the needed information, then click **Perform Quick Setup**.
   To review the current settings and note areas that needs correction, use the Results screen.

3.  For additional security options, use the following options.

    While not strictly a component of HTTP Basic Authentication, when HTTP Basic Authentication is enabled and active, additional security-related options can be configured:

    - Strict Passwords

        The administrator can require and configure the use of strict passwords. If strict passwords are enabled, the following criteria apply:

        – A password must contain at least eight characters.

        – A password must contain both letters and numbers.

        – A password must contain both lowercase and uppercase letters.

        – A password must contain at least one special character such as .#!$

   – A password must not contain the contextual information such as the user name or website name.

To enable requiring strict passwords, manually change vipo.ini, then restart the web Application. To enable strict passwords, use a text editor to add or change the line as follows:

```
vtpw_require_strict_passwords=1
```

A value of 0 indicates that strict passwords are not required.

- Password Reuse

Administrators can configure disallowing the reuse of passwords. If disallowing password reuse is enabled, users cannot use the same password more than once.

To disallow password reuse, manually change vipo.ini, then restart the web Application. To disallow password reuse, use a text editor to add or change the line as follows:

```
vtpw_disallow_password_reuse=1
```

A value of 0 indicates that password reuse is allowed.

- Login Session Expiration

Administrators can configure a login session expiration time out. If a user session is not active for longer than the specified time out, the session logs out and requires the user to log in again to resume the activity.

To enable login session expiration, manually change vipo.ini, then restart the web Application. To enable login session expiration, use a text editor to add or change the line as follows:

```
vtpw_session_maxduration_minutes=30
```

The number specified is the session expiration login time out value in minutes. A value of 0 indicates that the session never times out.

- Unsuccessful Login Attempt Lockout

Administrators can configure the number of times a user can unsuccessfully attempt to login before the account is locked.

To enable an unsuccessful login attempt lockout, manually change vipo.ini, then restart the web Application. To enable unsuccessful login attempt lockout, use a text editor to add or change the line as follows:

```
vtpw_unsuccess_login_max_attempt=5
```

The number specified is the maximum number of times a user can make unsuccessful attempts to log in before the account is locked. A value of 0 indicates that there is no limit on the number of unsuccessful login attempts.

- Account Lockout Duration

Administrators can configure how long a user waits after a lockout occurs before the user can attempt another log in.

To change the user wait time after a lockout, manually change vipo.ini, then restart the web Application. To change the duration, use a text editor to add or change the line. For example:

```
vtpw_account_lockout_duration_minutes=2
```

The number specified is the number of minutes a user is required to wait before attempting to log in again. A value of 0 indicates that there is no wait before the user is allowed to attempt a login.

**HTTPS (HTTP Secure) over SSL (Secure Sockets Layer)**

Beyond Basic Authentication, it can be useful to secure the communication channel between the client browser and the VIeC Web Application directly.

VIeC implementation of HTTPS (secure web Application)

1. The SSL Certificate needs to be installed from IIS.

2. To open the IIS Manager, in the Run command, type `inetmgr` or search for `Internet Information Service` in the **Start** menu.

3. In the Connections panel on the left side of the window, to change the port, navigate to **Server > Sites > Default Web Site**.

4. Click **FreeFlow VIeCompose**. On the right side of the window, you can find the **Browse Folder**.

5. To launch Web Application, click **Browse *.80(http)**, or to launch Secure Web Application, click **Browse *.443 (https)**.

6. By default, there is a self-signed SSL certificate and that certificate needs to be changed to the correct SSL certification.

7. From the client browser, replace http in the VIeC Web browser URL with https:
   `https://url_name`

   or

   `https://nn.n.n.nnn`, where nn.n.n.nnn is the IP address.


**SSL Certificate Installation**

The following instructions guide you through the SSL installation process on Microsoft IIS 7. If you have more than one server or device, you need to install the certificate on each server or device that you need to secure.

To install the SSL certificate, refer to the following instructions:

1. Launch IIS Manager. Click **Start > Control Panel > Administrative Tools > Internet Information Services (IIS) Manager**.

2. Select your server name. In the left Connections menu, select the server name as host where you want to install the certificate.

3. Navigate to the Security section. In the center menu, click the **Server Certificates** under the Security section near the bottom.

4. Click **Complete Certificate Request**. In the right **Actions** menu, click **Complete Certificate Request**.

5. Browse to your Server Certificate. In the Complete Certificate Request wizard field, to browse and select your Server Certificate file that you saved on the desktop of your server, click **…**.

6. Name your certificate. Enter a Friendly Name, which is an internal reference name that is used to distinguish the file later. It is recommended that you include the Certification Authorities name and expiration date.

7. Click **OK**.

   The installed certificate appears in the refreshed Server Certificate List.

**Binding your Certificate to your Website**

Proceed with the following steps, which help you assign or bind the SSL certificate to the appropriate website.

1.  Access your Sites folder. From the left Connections menu, expand the name of your server, expand the **Sites** folder, then expand the **Default Web Site** folder.

2.  In the right Actions menu, click **Bindings**.

3.  In Site Bindings window, click **Add**.

    🖉 Note: If you already have the appropriate site binding created, click **Edit** then change the SSL Certificate accordingly.

4.  In Add Site Bindings window, enter the following information:

    • For Type, select **HTTPS**

    • For IP Address, select **All Unassigned**. If you have multiple IP addresses, select the IP address that applies.

    • For Port, enter 443 unless SSL traffic appears on another port.

    • For SSL Certificates, select the friendly name of the SSL certificate that you installed. To confirm the validity details of the certificate, click **View**.

    • To finish binding the SSL certificate to your live website, click **OK**.

    🖉 Note: For changes to take effect, you need to restart your server.

**OpenSSL License Statement**

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit, http://www.openssl.org/.

## VIeC Server Node Installation

This is an advanced option for use by the VIeC system administrator. Before you install a VIeC Server, you must:

- Obtain a license for each VIeC Server

- Map a network drive (for example, X) to the shared drive on the file server

On each workstation on which a VIeC Server has to be installed, select **VIeC Server Node servicing a VIeC Cluster**.

When prompted, choose the network drive of the file server that has been set up for use by the VIeC Cluster.

During installation, for each workstation on drive C, the user_root is set automatically to:

```
user_root=X:\Program Files(x86)\Xerox\VIPP\xvtp\users
```

This indicates that each VIeC Server treats jobs placed on the file server as if they were local; that is, it processes jobs on the file server and places the output on the file server.

After bringing up the first VIeC Server, you will have a configuration similar to this:
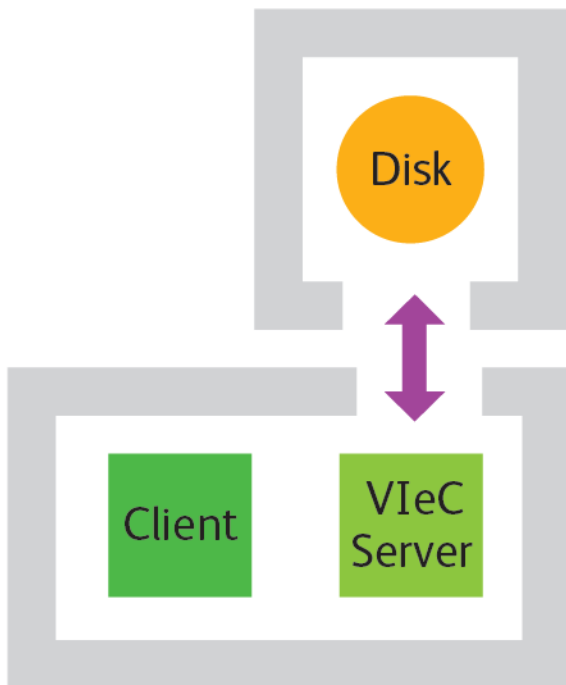


To test the installation, submit a small local job from the Job Submission Client on the VIeC Server. The job results are generated on the file server, not on the VIeC Server workstation.
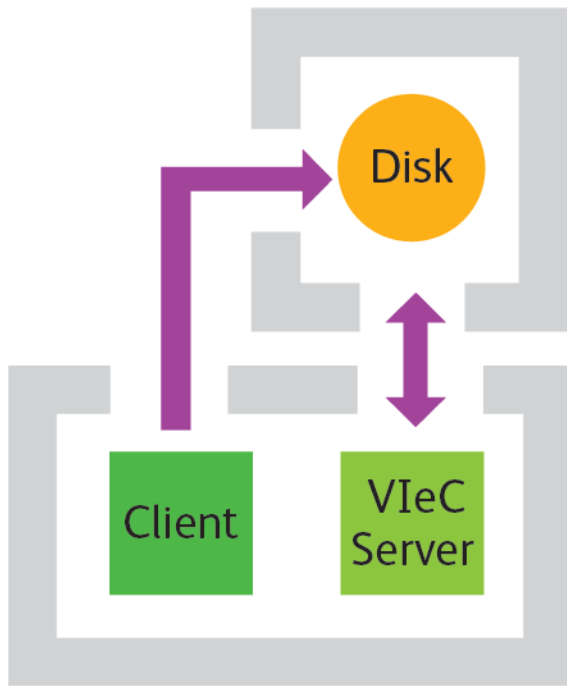
More VIeC Servers can be added to the Cluster by repeating this installation process.
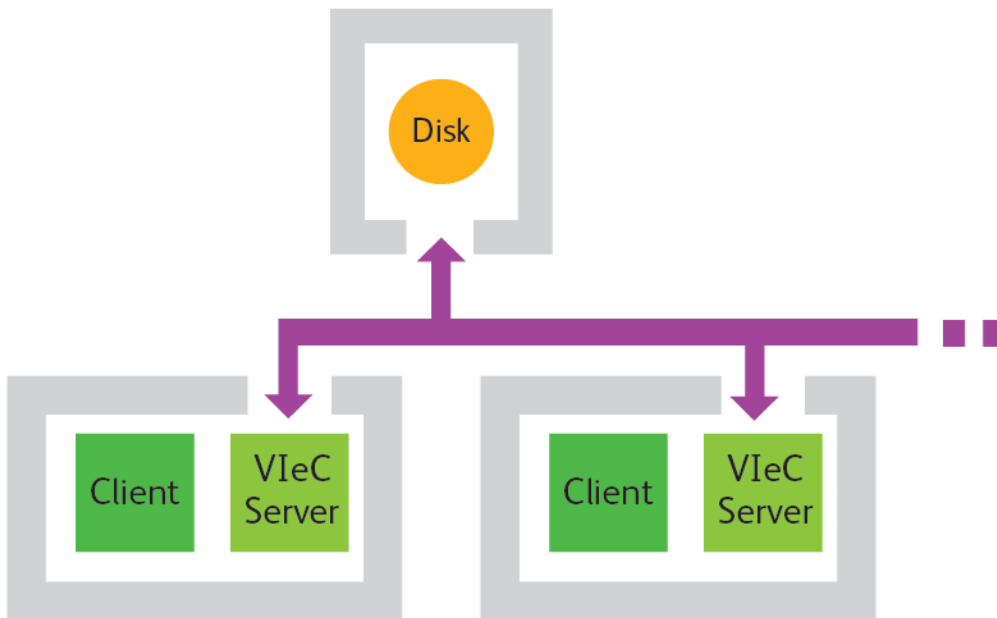


During the installation process, a skeleton directory structure is installed on the file server, which makes it appear to VIeC Clients as a VIeC Server. This allows VIeC Clients to submit jobs to the file server as if they were a VIeC Server. Now, Client-only installations can be pointed at the file server to set up a VIeC Cluster.

VIeC Server Nodes in a Cluster cooperate transparently when selecting jobs, that is they do not get in each other's way. You can add Server Nodes to the Cluster to improve the job turnaround time or remove them for maintenance while the Cluster itself is active, as long as at least one VIeC Server Node is servicing the Cluster. Before you remove

a node, ensure that you shut down the node gracefully. The only impact on clients is in the improvement or degradation of their job turnaround time.
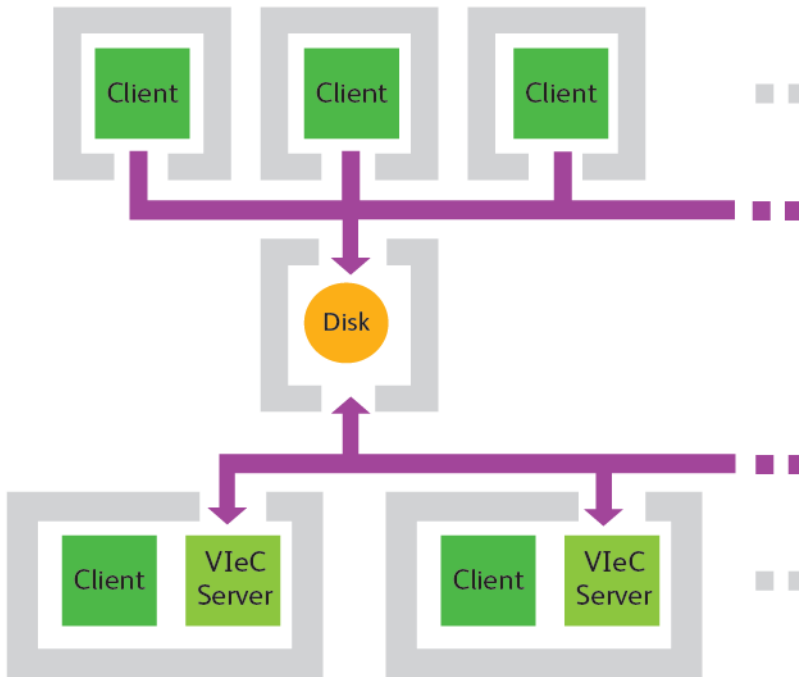


### POST-INSTALLATION STEPS

When you configure Xerox® FreeFlow® VI eCompose in a cluster configuration, to grant access to remote resources for the default_user, perform the following. Remote resources include access to remote files using mapped network drives.

- If all the remote resources are located in the same domain, and if the logged-in user has the read and write access to the specific domain location, then the application has access to those locations by default.

- If the remote resources are located in different networks or if the resources require different credentials or non domain user account, enable the remember credentials option to store the credentials while mapping the drive in Windows and perform the following steps:

1. Open the IIS Management Console (INETMGR.MSC).

2. Open the Application Pools node underneath the machine node. Select the **WebAppPool** application.

3. Right click the **WebAppPool** application pool, then select **Advanced Settings**.

4. Select the Identity list item and click the ellipsis (the button with the three dots).

5. Select the **Custom Account** radio button, and click **Set**. Enter the name and password of your current administrator account, and click **OK**.

6. Click **OK**.

7. Restart IIS and refresh VIeC UI.

   Server instance of the cluster is displayed.

## VIeC Web Job Submission Service (WJSS) Installation

To run VIeC Web Job Submission Service, you must have the following installed software:

- FreeFlow VI eCompose, version 16.0, or later. If this is a new install, reboot the system before installing VIeC Web Job Submission.

- Microsoft .NET Framework 4.6.1

WJSS can be hosted as a Windows Service or by Internet Information Services (IIS). To host WJSS by IIS, the following software is required:

- Internet Information Services (IIS) 7

- Microsoft Web Deploy (3.0 or later), an extension of IIS

The following files are required. They are included in the VIeC installation zip file, under the `..\..\WJSS` folder:

- WJSSProxyServer.zip

- WJSSTestClient.zip

Microsoft Visual Studio 2013/17, for modifications to the WJSS Test Client, is optional software.

> 🖉 Note: When installing VIeC, be sure to select the Client and Server configuration (first option) and follow steps to complete installation. Reboot the system when requested.

For further WJSS installation instructions, refer to `VIeComposeWJSS.pdf` in the `..\..\WJSS` folder.

## VIeC Services Installation

Before you begin installing VIeC Services:

- To verify that the Local Server and the Job Submission Client are functioning properly, on the local printer with GUI consoles, test the latest VIeC setup.

- Procedures and screens can vary. The screenshots and procedures in this section are based on a Windows 7 system. When installing on a different operating system, the procedures and screens can vary.

VI eCompose Services are intended to run as Windows Services. While the service component files are installed physically in the bin directory as part of the VI eCompose installation process, the service component files are not installed as Windows Services. Your system administrator can install the files on a case-by-case basis. Familiarity with the Windows Services administration is recommended for anyone wanting to install, run, or uninstall one or more of the VI eCompose Services.

The VI eCompose Services and their associated executable files are as follows:

| FreeFlow VI eCompose Service | vtpsrvc.exe |
|---|---|
| FreeFlow VI eCompose Watched Folders Service | vtpfsrvc.exe |
| FreeFlow VI eCompose Dispatch Service | vtpdsrvc.exe |
| FreeFlow VI eCompose Web Job Submission Service | vinipmsrvc.exe |

To install or uninstall the services on Windows® 10, Windows® 11, Windows Server® 2016, Windows Server® 2019, and Windows Server® 2022, two Java Scripts are included with the VI eCompose installation:

- InstallServices.js
- UnInstallServices.js
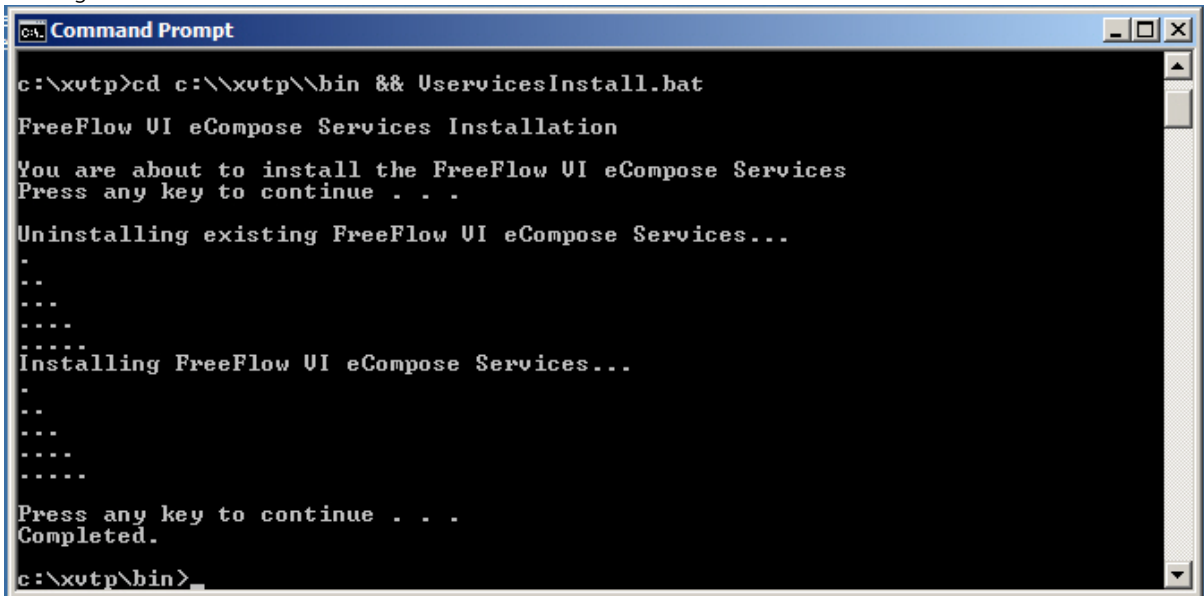
To execute the scripts, go to the `X:\Program Files(x86)\Xerox\VIPP\xvtp\bin` directory, where X is the drive where VIeC is installed. To install or uninstall the VIeC services, double-click the appropriate file. Elevated system administrator privileges are required to install and uninstall services on Windows® 10, Windows® 11, Windows Server® 2016, Windows Server® 2019, and Windows Server® 2022.

### INSTALLATION PROCEDURES

Follow these steps to install VI eCompose Services:

1. Ensure that no VI eCompose clients, servers, or services are running.

2. From a DOS command prompt, change to the `X:\Program Files(x86)\Xerox\VIPP\xvtp\bin` directory, where X is the partition where the VI eCompose software is installed.

3. At the prompt, enter the command `InstallServices` or `InstallServices.js`. The system uninstalls existing VIeC services and installs the new version as shown:



4. Optional, but recommended: From the Windows Startup folder at `C:\ProgramData\Microsoft \Windows\Start Menu\Programs\Startup`, remove or delete the shortcut for `VI eCompose Local Server`.

After you are finished, to complete the installation, follow the instructions in the following sections:

- Configure automatic startup for local access

- Confirm automatic startup for local access

- Edit the.ini file for remote access

- Add a new user for remote access

- Configure automatic startup for remote access

- Confirm automatic startup for remote access

**Configure Automatic Startup for Local Access**

Use these steps to configure the desired VI eCompose Services to start automatically at system boot-up.

1. From the Windows Control Panel, navigate to **Administrative Tools**, then select **Services**. This can be at **Control Panel > Administrative Tools > Services**, or as follows, at **Control Panel > System and Security > Administrative Tools > Services**.
Scroll down in the Services dialog until you see the four VI eCompose services. These services are initially set to start manually and to use the Local System account. Select the services that you want to start automatically.

2. Right-click the **VI eCompose** service, then select Properties from the menu. A dialog window appears.

3. Change the Startup type to **Automatic**. After you change the startup type, click **Apply** and then click **OK**.

**Confirm Automatic Startup for Local Access**

Restart the local machine and examine the **Task Manager** to ensure that the desired VIeC services are running automatically. Test again with GUI consoles, such as Job Submission Client, to verify that they are working properly.

**Edit the .ini File for Remote Access**

✎ Note: This step is necessary only if the drive that contains the user root is a network or mapped drive. If the user root drive is on the same physical printer, you can skip this step.

After the VIeC Service is installed and verified on the local printer, manually change the `vipo.ini` file. Use a text editor to change the line:`user_root=X:\Program Files(x86)\Xerox\VIPP\xvtp\users` to `user_root=\\SERVERNAME\USERS`, where X is the partition where the VI eCompose software is installed.

Save the `.ini` file in the original location.

- SERVERNAME and USERS: In the `.ini` file, `SERVERNAME` is the Network Computer Name ID of the system, and `USERS` is a user-defined folder name ID that maps to the VIeC root user folder path `X:\Program Files(x86)\Xerox\VIPP\xvtp\users`.

- Make the USERS folder a shared folder, then set up the proper read and write access permissions.

- For remote access: For mapped or shared drives, it is recommended that you use UNC paths, for example: `\\SERVERNAME\USERS`. Otherwise, for local access, continue to use designated drive letters and local paths, for example: `C:\Program Files(x86)\Xerox\VIPP\xvtp\users`.

**Add a New User for Remote Access**

1. Navigate to the panel where user accounts are managed for your operating system. This can be at **Control Panel > Users and Passwords, Control Panel > User Accounts**; or as follows, at **Control Panel > User Accounts and Family Safety > User Accounts > Manage Accounts**.

2. Create a new login account by clicking the **Create a New Account** link or **Add**, then follow the instructions in the dialog windows that appear.
   In addition to any basic information necessary to create a new user, ensure to enter login and password information in addition to assigning Administrator privileges to the account.

   When the new user is created, a window appears with the details of the new account created.

**Configure Automatic Startup for Remote Access**

First configure automatic startup for local access as explained in Configure Automatic Startup for Local Access. Then reconfigure the installation to use the VI eCompose Services account:

1. From the Windows Control Panel, navigate to **Administrative Tools** and select **Services**. This can be at **Control Panel > Administrative Tools > Services**, or as follows, at **Control Panel > System and Security > Administrative Tools > Services**.
   Scroll down in the Services dialog until you see the four VI eCompose services. Right-click on a VIeC service, then select **Properties** from the menu.

2. Click **Log On** tab of the Properties dialog window.

3. From the Log On dialog, select **This account**, then click **Browse**.

4.  When the system produces a list of available accounts, highlight **VI PDF Services**, then click **OK**. A new screen appears.

5.  If your operating system displays a screen that looks like the password is preset, it is not. You must enter a new password, confirm the password, then click **OK**.

6.  If the screen from step 4 appears, select the correct location, then select the user name using the **Check Names** option.

7.  After you enter the password, VI eCompose Dispatch Service is associated with the VI PDF Services account. When a dialog window appears that confirms the change, click **OK**.

    The system returns to the Services window, that shows that VI eCompose Dispatch Service has a Startup Type of Automatic and a Log On As of .\VI PDF Services.

8.  Repeat the steps in this section for each VI eCompose component service that will automatically run at system boot up. Ensure that all VI eCompose component services are associated with the .VI PDF Services log on.

**Confirm Automatic Startup for Remote Access**

VI eCompose services are installed now and configured but are not running. You can start the services manually.
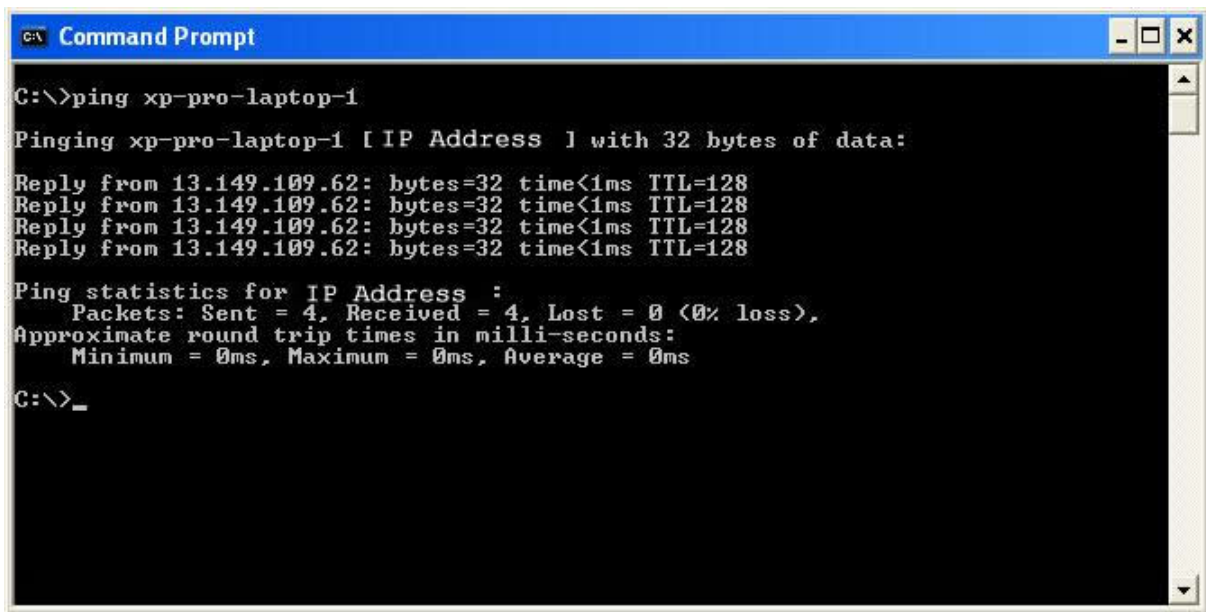
Follow these steps to confirm that the system can operate as a true hands-free VI eCompose services device to start up automatically.

1.  Close the Services dialog, then restart the PC. After the PC restarts, do not log in. Leave the PC at the desktop login dialog.

    Note: After the login dialog appears, it can take VI eCompose components an additional minute or two to completely start up and synchronize the operations.

2. After waiting 1–2 minutes, go to a different device on the network, then ping the device that is running the VI eCompose components. The information consists of specific reply information, ping statistics, and approximate round-trip times.



If you get timeouts, there can be a software firewall that runs on the VI eCompose component device, or some other security barrier is in place. Before you continue, resolve the issues, then get a successful ping.

3. Log in remotely to the VI eCompose Server. This action is possible when the VI eCompose component device is running VI eCompose Web server as a Web service. Log in remotely using the servername/FreeFlowVIeCompose/ or the VIeC Server address/FreeFlowVIeCompose/ of the VI eCompose component device, depending on how the network is set up.

4. As a test, submit a sample job to VI eCompose. The example shows the submission of the `billb.lm` job. The example shows that the VI eCompose component device accepted and processed job without anyone being physically logged in to that device.

When someone logs into the device where VI eCompose services are running, and the account has VI eCompose components installed to start automatically at login, one or more dialog boxes appear, indicating that a VI eCompose server is already running as a Windows Service. This action is normal. Only one instance of a VI eCompose server can run on the device at a time, and since there is already one running as a service, another instance cannot be started from the desktop.

To use a VI eCompose component in normal GUI mode, go back to the Services dialog, then use the right-click menu on the corresponding service. Select **Stop**.

After the service has stopped, start the VI eCompose component in GUI mode. For example, click **Start > All Programs > Xerox FreeFlow VI eCompose xx > VI eCompose Local Server**.

🖉 Note: When a VI eCompose component is running in normal GUI mode, and an attempt to run the component as a service is made, an error message is produced. In this example, only one instance is allowed, and the Services dialog produces an error message. To resume operating the component as a service, shut down the normal instance of the server.

**Setup Multiple Instances of VIeC Server and Dispatch as Services**

When starting as Windows services, VIeC Server and VIeC Dispatch modules look for the following entries in the `.ini` configuration file:

- `vtpserver_number_instances=S`
- `vtpdispatch_number_instances=D`

The S and D are integers that determine how many instances of VIeC Server and VIeC Dispatch are launched when running as Windows Services. Note that these entries are relevant only for Services startup in the multi-instance mode. The entries are ignored when VIeC Server and VIeC Dispatch are started as standard GUI applications from the **Start > All Programs** menu.

**Requirements to Run Multiple Instances of VIeC**

For best results, a minimum of one CPU or core and 2 Gbytes of RAM is required for each instance of VIeC. If Dispatch is involved, enough CPU memory is required for all simultaneous post-processing operations, which vary based on the load and complexity of the jobs to be processed.

A minimum configuration for four instances. without taking Dispatch into account, is a quad core system with at least 8 Gbytes of RAM, running on a 64-bit operating system. If there is Dispatch post-processing or other system loads, more CPU and RAM can be added as required at the site.

**STOREVAR** command error

**Error**

    VIPP_invalid_combination; OffendingCommand: STOREVAR_VIPOmulti

The `VIPP_invalid_combination` error is raised when the **STOREVAR VIPP** command is used in a job submitted to VIeC that is running in multi-instance mode. The main purpose of **STOREVAR** is to allow rolling numbers to pass from one job to the next in a sequential fashion, but multiple VIPP® jobs running in parallel defeats the intended goal.

The use of the **STOREVAR VIPP** command in your production jobs on VIeC software depends upon your knowledge of how **STOREVAR** is used.

- If **STOREVAR** is used in a way that forbids multiple parallel instances, for example, rolling numbers, terminate all running VIeC modules. Reset the `.ini` entry `enable_multi_instance=0`.
  - This action disables the multi-instance mode for VIeC Server and Dispatch, which prevents the **STOREVAR** error. :
  - To run VIeC again in multi-instance mode, set the same `.ini` entry to `enable_multi_instance=1`. You can run multiple instances of VIeC Server and Dispatch on the same device.
- If **STOREVAR** is used in a way that allows multiple parallel instances, you can use the following syntax for **STOREVAR**:
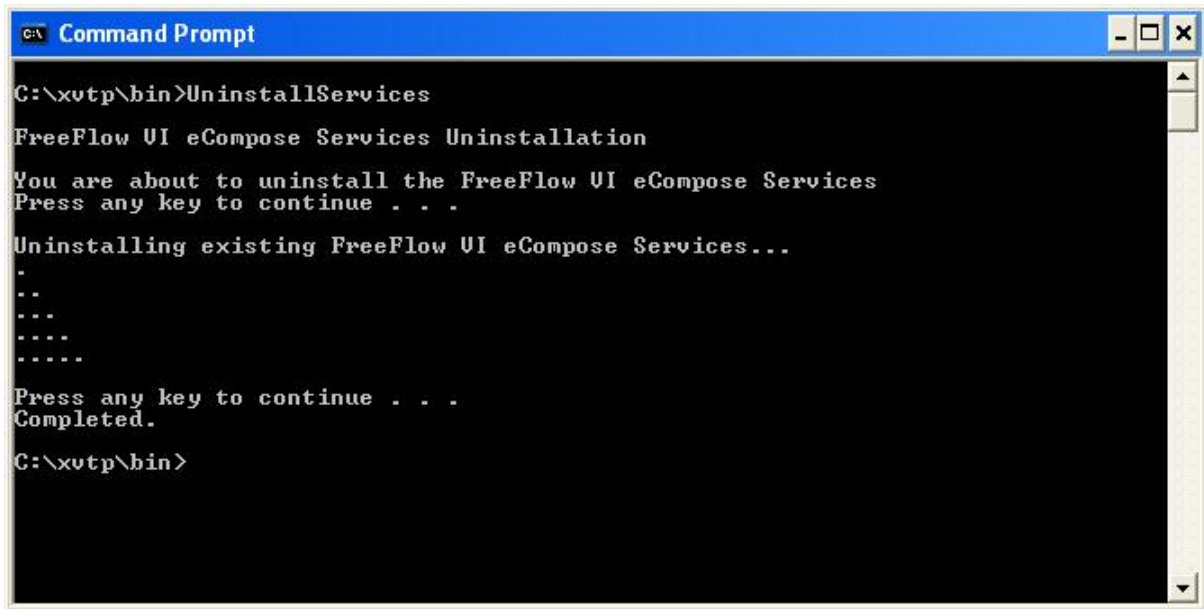
    `data (file name) /M STOREVAR`

  The /M switch disables the **STOREVAR** error detection when VIeC is run in multi-instances mode. Note that when this switch is used, if two jobs are running in parallel, and **STOREVAR** executes with the same file name, an error occurs, or the final content of the file referenced by **STOREVAR** can be incorrect.

## UNINSTALLING VI ECOMPOSE SERVICES

Follow these steps to uninstall VI eCompose Services:

1. Ensure that no VI eCompose servers or services are running.

2. Run the **UninstallServices** or **UninstallServices.js** batch file from the `\xvtp\bin` directory:

## Adobe Acrobat and Reader

If you are using PDF files as VIPP® resources in your VIPP® application and you need to view the PDF file output from VIeC using Adobe Acrobat or Reader, ensure that you are using Adobe Acrobat 10.1.6 or later or Reader 10.1.6 or later.

To view PDF files that contain VIPP® PDF resources, the PDF files must be opened from an Adobe Acrobat or Adobe Reader privileged folder. To use an Adobe Acrobat or Adobe Reader privileged folder, create the folder, then set enhanced security options for that folder. The steps to complete that process follow.

### CREATE AN ADOBE ACROBAT OR READER PRIVILEGED FOLDER

To create an Adobe Acrobat or Reader privileged folder, follow these steps:

1. In Acrobat or Reader, select **Edit > Preferences**, or to open the Preferences folder, press **Ctrl+K**.

2. In the Categories menu, select **Page Display**.

3. From the Show reference XObject targets list, select **Always**.

4. To insert the directory containing the PDFs generated by VIeC in the Location of referenced files field, select **Browse for Location**.

    Note:

    • It is recommended that you select the VIeC incoming folder instead of the folder containing a specific job.

    • As long as all of the PDFs generated by VIeC are under the same incoming folder hierarchy, all referenced PDFs can be found and properly displayed.

### ADJUST SECURITY SETTINGS

To adjust Security settings, follow these steps:

1. If the Preferences Panel is not already open, select **Edit > Preferences**, or press **Ctrl+K**.

2. In the Categories menu, select **Security (Enhanced)**.

3. Ensure that Enable Enhanced Security is selected.

4. Click **Add Folder Path**, then specify the same folder as specified for the location of the referenced files.

    Note: Referenced XObjects are independent of the Enhanced Security check box. The state of the check box does not govern viewing of referenced XObjects.

Any of the following states can cause Acrobat or Reader to display `Missing:`, followed by the name of the PDF resource:

• The version of Acrobat or Reader does not support Referenced XObjects.

• The preferences are set up incorrectly.

• The PDF file was opened from a non-privileged folder.

# VI eCompose Overview

This chapter contains:

## VI Suite Customer Forum

Xerox hosts a Community Support Forum. The VI Suite Customer forum is now part of this larger support forum, allowing you to post and review information about Xerox products and services all from one location. Please take a minute to log into this customer forum community: http://vippsupport.xerox.com.

## VIeC

VIeC is a client/server application that allows you to generate Adobe PDF documents from VIPP®-based variable data applications and forward the applications to other processes within the environment.

VIeC extends the VIPP® workflow into electronic distribution and archive by providing the ability to generate Adobe PDF files from the same data files sent to a VIPP®-enabled print device. The PDF files, along with information from the data record that produced the PDF files, can then be passed to a user-defined process using VIeC Dispatch. The files can be integrated into processes within the environment, which can include email servers or archive systems. In addition, the VIeC Server can forward the data submission file or the Master PDF file to an identified VIPP®-enabled print device, or for the Master PDF, to a print device available to the Printer dialog box on the Windows server for hard copy output.

A standard VIPP® application that is submitted to a VIPP®-enabled print device can be submitted directly to the VI eCompose server. No changes in the data or VIPP® resources are required. VIPP® resources are installed on the VIeC Server just as they are installed on a printer disk. VIeC uses the same resource management structures.

VIeC operates like any other VIPP®-enabled printer. A job submitted to the print device using lpr can use the VIeC lpr client to submit to VIeC. In addition, VIeC provides other job submission clients:

- Watched Folder client

- Web submission client

- Job Submission client

- VIeC Web Job Submission Service (WJSS)

  🖉 Note: WJSS is a web-based submission method to enable faster production of jobs through the VIeC Server. Although WJSS is often used for small jobs, WJSS is not limited by the size of the job. Using WJSS bypasses the standard VI eCompose processing model to provide faster job turnover by improving the latency between processed jobs. For more information, refer to VIeC Web Job Submission Service.

A VIPP® job processed by the VIeC Server results in the generation of an Adobe PDF file. This PDF file is output to a named directory on the server. You can specify a print path. The data file processed by VIeC, or the Master PDF file, is forwarded to the print device after the data file is processed by VIeC. This action allows both electronic and hard copy printing from a single job submission.

Any job that uses the VIPP® **BOOKMARK** command can be enhanced to create individual sets of PDF files. For example, a statement run that contains 10,000 customer records can be split into 10,000 individual variable-length customer PDF statements. These PDF files, along with additional information contained in the bookmark, can be dispatched to a user-defined process that can include, but is not limited to, web presentment, archive, or viewing systems, all without making any changes to the data or the workflow.

PDF Interactive features such as PDF Bookmarks, PDF notes, or URLs, can be added to the PDF output file using the VIPP® **SETPIF** command. In the statement customer records example, several additional PDF bookmarks can be set to guide the user to different sections within the statement. You can place PDF notes in the PDF file and insert a URL into the PDF that can link the user to additional web sites.

The built-in variable *VARDataFileName* is set when the job is processed by VIeC. The variable contains the name of the data file when submitted directly, or the name of the submission data file specified in the VI Project Container that gets submitted for processing.

For more information about VIeC, refer to Using VIeC.

## VIeC Support for FreeFlow Process Manager

VIeC has been modified to support a simple, but limited, interface with FreeFlow Process Manager 7.0 or later. This interface is designed to provide a hands-free method to process a VIPP® application and deliver a single PDF file into the Process Manager workflow.

### WORKFLOW

The FreeFlow Process Manager/VIeC integration workflow is described below.

1. A VIPP® VI Container file, `.vpc`, that contains all the VIPP® resources and data submission file is passed to the Process Manager. The Process Manager verifies that the `.vpc` file is a VIPP® VI container file. If it is of another type, the Process Manager raises an error. If the VI Container is a valid VIPP® VI Container, the Process Manager posts the `.vpc` file to a watched folder that VIeC is watching.

2. VIeC processes the VIPP® VI Container file and passes a status to the Process Manager. If processing completes without an error, the PDF file is passed to the watched folder that the Process Manager is watching.

3. VIeC performs file clean-up, deleting the VI container file and other files as required.

4. The Process Manager takes control of the PDF file and performs the next action in the Process Manager workflow.

### LIMITATIONS

This special FreeFlow Process Manager and VIeC integration is designed to provide an easy to configure and easy to use pathway to process VIPP® applications, then pass the resulting single PDF file into the Process Manager workflow. Because of this ease of configuration and ease of use, many of the advanced features of VIeC are not available when installed in this manner. For example, none of the available clients are accessible, dispatch is not accessible, and distill options such as Distill and Split are not accessible. The FreeFlow Process Manager and VIeC configuration allows the user to pass only a VIPP® `.vpc` file to the Process Manager and have VIeC process the file and pass back the single PDF file to Process Manager automatically.

If you require the full functionality of VIeC, select the Standard installation process. The Standard installation process installs VIeC using the standard package and allows access to all the clients, features, and the dispatch module. If required, you can configure Process Manager and VIeC to work together using Watched Folders and VIeC Dispatch. For more information refer to FreeFlow VI eCompose Installation.

## SAP Device Type for VI Compose

The Xerox FreeFlow VI Systems, Applications and Products in Data Processing (SAP) Device Type and Form provides functionality and feature integration with the FreeFlow VI Compose and VI eCompose products. Although SAP provides support for many output devices, this support does not include the VI enabled printers and VI enabled eCompose Servers (VIPP® to PDF).

The FreeFlow VI SAP Solution allows SAP users to access the more advanced feature set of the VIPP® scripting language, providing a fast, efficient, effective, and flexible method for printing or creating Adobe PDF files from a delimited XML or line data (print ready data) files.

The sample device type and form are provided in a zip file, `VISAPDT.ZIP`, that can be downloaded from www.xerox.com. Go to the Support & Drivers webpage and search for VIPP®, then select the Software option. The zip file is found in the Utilities & Applications section. Ensure that the contents of the zip file are reviewed by the SAP systems administrator.

SAP uses the legacy VIPP® path using the Just Send the Data model. VIPP® resources, such as fonts, images, and forms, must be uploaded to the printer `XGFC` folders. Uploading can be done during the development of the VIPP® application. When the resources are available to the print device or the FreeFlow VI eCompose server, the SAP print job can be processed using the SAP Device Type that is configured with the VIPP® start commands. In effect, this process is sending the data to the target device.

The only requirement is that FreeFlow VI Compose is installed and licensed on the target print device. If the SAP administrator needs to output PDF, those VIPP® jobs can be sent to the FreeFlow VI eCompose server.

## VIeC Dispatch

VIeC Dispatch (VIeCD) is the generic mechanism by which VIeC jobs that have been distilled and split, or split into pages, are sent, or dispatched, to other programs or functions such as:

- Email

- Fax

- DocuShare repository

VIeCD monitors VIeC output and performs post-processing using a customer-specified back-end. Post-processing is completed in accordance with parameters and other data embedded in a job. Post-processing is communicated to the dispatch mechanism using the index file, specifically the splitfile index record field names and values, of completed VIeC jobs.

Because different applications require different workflows, VIeCD handles these types of workflow simultaneously:

- Jobs in which the entire VIeC-to-VIeCD process runs automatically, such as an interface with a DocuShare repository.

- Jobs that require verification of VIeC output before the dispatch proceeds, such as an interface with an email disbursement system.

VIeCD inspects, or polls, the VIeC Incoming Folders directories in the context of a user-specified filter and compares the field names of the first record from the splitfile index of the VIeC job against a repository of dispatch rules. If a completed VIeC job is associated with a single dispatch rule, it is considered eligible for dispatch and becomes a VIeCD job. If a VPTD job is either manually approved for processing or meets the user-specified criteria for autorun, from another filter, VIeCD resolves and applies the appropriate dispatch rule for each record of the splitfile index of the VIeC job .

Each of the splitfile records of the VIeCD job in the context of a dispatch rule effectively defines a subtask of the VIeCD job. Processing of each subtask first, optionally writes the contents of one or more splitfile index records to disk and then performs a single invocation of a rule-specified external program. For example, an email client interface. The results of subtask processing are accumulated in a log file in the Incoming directory of the VIeC job and represent the results of the overall VIeCD job.

This VIeC component is discussed in detail in Using VIeC Dispatch.

## VIeC Services

VIeC and VIeC Servers can be initiated from a Windows Service, which allows the various servers to be initiated and to perform operations without requiring a manual logon. Launching and activating VIeC and VIeC servers when the system is booted up allows a hands-off VIeC system. Use of VIeC Services depends entirely upon the proper installation of the functionality. For a complete description of the installation process, and related background information, refer to VIeC Services Installation.

A VI eCompose Server is one of the following VI eCompose components:

- FreeFlow VI eCompose

- FreeFlow VI eCompose Watched Folders Client

- FreeFlow VI eCompose Dispatch

- FreeFlow VI eCompose Secure Web Application

- FreeFlow VI eCompose Web Application

These components, once running, perform various operations related to VI eCompose.

This VIeC component is discussed in detail in Using VIeC Services.

### SETTING UP AND DEBUGGING VI ECOMPOSE SERVICES

Before setting up VI eCompose Services, it is recommended that you configure and test each individual VI eCompose module as a GUI application so that you can verify the workflow and see the job processing in action. After you configure and test each VI eCompose module, start a VI eCompose Service, one at a time, and verify the workflow again as each service starts. If there is a problem, stop VI eCompose services one at a time in the reverse order of startup, then check if the problem goes away when a particular VI eCompose Service stops.

> Note: While all VI eCompose Services are installed with the batch or script file provided, not all Services need to be used or even started. Each customer requirement is different, so customize the VI eCompose configuration as needed.

When you use VI eCompose Services on systems that access remote file servers on network-mounted drives or shared folders, it is recommended that you modify the appropriate `.ini` options using a `UNC` path format. For example, to access the remote folder `X:\Program Files(x86)\Xerox\VIPP\xvtp\users` mapped to a shared folder named `VIPOUSERS` on a file server named MyFileServer, the UNC path is:

`user_root=\\MyFileServer\VIPOUSERS`

> Note: It is critical to ensure that any file system that is accessed remotely has the proper read-write permissions for the user account that runs VI eCompose Services. Failure to set up the user account with the appropriate read-write permissions causes jobs to fail.

## VIeC Web

VIeC Web allows you to use the web browser to:

- Submit VIPP® jobs

- Create new or additional VIeC user accounts

- Administer VIeC Servers remotely

- Administer VIeC Web passwords and other security functions This VIeC component is discussed in detail in Using VIeC Web.

## VIeC Configuration Options

VIeC allows you to increase the performance of the VIeC environment in several ways:

- Upgrading the server running VIeC to a multi-core platform and increasing the memory

- Multiple VIeC systems can be run in a cluster configuration in which all the servers act together to process available jobs

- Multiple instances of VIeC can be enabled and run on a single server

With a multi-core server, you will see improvements using any one, or a combination of the listed configurations. It is acceptable to have multiple servers in a cluster configuration, each running multiple instances of the VIeC engine.

The configuration options available to the site are listed in these sections:

- VIeC Multiple Instance

- VIeC Server and Client

- VIeC Client only

- VIeC Server Node servicing a VIeC Cluster

### VIEC MULTIPLE INSTANCE

You can run multiple concurrent instances of VIeC and VIeC Dispatch on a single platform. This option can be run as often as needed, however, it is limited by the capabilities of the computer on which VIeC is installed. The two types of multiple instance support are as follows:

- Multiple VIeC Instance Support

  VI eCompose (VIeC) has been modified to support multiple concurrent instances of the VIeC server on a single platform. End-to-end processing time of individual jobs is not be reduced by this new feature, however in the presence of multiple jobs, this feature enables more than one job to be picked up and processed at a time. This action can improve the VIeC server throughput performance significantly when running multiple jobs on a server that has multiple CPUs installed. Users supply suitable hardware to take full advantage of these changes. This implementation does not require multiple processors to run. When using a server that supports multiple processors, greater throughput performance benefits are seen.

- Multiple Dispatch Instance Support

  VI eCompose has been modified to support multiple instances of the Dispatch module. This modification allows multiple instances of Dispatch to process jobs simultaneously. As with multiple VIeC instance support, the end-to-end processing time of individual jobs is not reduced by this new feature, however in the presence of multiple jobs, this feature enables more than one job to be picked up and processed at a time.

When the server is started after installing VIeC, a single instance of VIeC is started. You can start additional instances of VIeC. From the VIeC Program Group in Windows, click **Start > All Programs**, then start the VI eCompose Local Server option. There are two ways to start instances of VIeC Dispatch:

- You can start instances of VIeC Dispatch in a similar manner to starting additional instances of VIeC.

- From the task bar, click the **VIeC** icon, then select **Launch Dispatch**.

An icon appears in the task bar for each running instance of VIeC or Dispatch.

✎ Note:

- Only one Manual Job Submission panel can be displayed, no matter how many instances of VIeC are active.

- Multiple instances and billing information:

  – When multiple instances are running, billing information is derived from the contents of the accounting logs in the same manner as when running multiple VIeC nodes in a VIeC cluster.

  – The Billing Meter reading is accurate only when one instance of VIeC is running.

Unlike nodes in a VIeC Cluster in which only a common job pool is shared, multiple VIeC/VIeCD instances share the job pool and accounting and log files, profiles, security options, distiller job options, and any dispatch filters and rules on that particular system. Be careful when you configure, edit, or otherwise change settings that can be accessed by the running instances of either VIeC or VIeC Dispatch on a given system. It is recommended that only one VIeC/VIeCD instance runs on a given system when configuring or setting up for production, and that multiple instances are started only when the production configuration is deemed ready.

The number of instances of VIeC/VIeCD to run depends on the capability of the hardware, the number of CPUs and amount of RAM, and the specific workflow. It is recommended that one instance of a server is launched for each CPU or core and 2 Gbytes of RAM on the system. It is recommended to launch one instance of a server from a logged in account in interactive mode, not running the instances as a service. From the logged in account, evaluate performance, throughput, and the number of instances adjusted.

**Multiple Instances of VIeC Server and Dispatch as Services**

When starting as Windows services, VIeC Server and VIeC Dispatch modules look for the following entries in the `.ini` configuration file:

- `vtpserver_number_instances=S`

- `vtpdispatch_number_instances=D`

The S and D are integers that determine how many instances of VIeC Server and VIeC Dispatch are launched when running as Windows Services. Note that these entries are relevant only for Services startup in the multi-instance mode. The entries are ignored when VIeC Server and VIeC Dispatch are started as standard GUI applications from the **Start > All Programs** menu.

**STOREVAR** command error

**Error**

```
VIPP_invalid_combination; OffendingCommand: STOREVAR_VIPOmulti
```

The `VIPP_invalid_combination` error is raised when the **STOREVAR VIPP** command is used in a job submitted to VIeC that is running in multi-instance mode. The main purpose of **STOREVAR** is to allow rolling numbers to pass from one job to the next in a sequential fashion, but multiple VIPP® jobs running in parallel defeats the intended goal.

The use of the **STOREVAR VIPP** command in your production jobs on VIeC software depends upon your knowledge of how **STOREVAR** is used.

- If **STOREVAR** is used in a way that forbids multiple parallel instances, for example, with rolling numbers, after you terminate all the running VIeC modules, reset the `.ini enable_multi_instance=0`

- The command disables the multiple-instance mode for VIeC Server and Dispatch, which prevents the STOREVAR error.

- To run VIeC again in multiple-instance mode, set the same `.ini` entry to `enable_multi_instance= 1`. You can run multiple instances of VIeC Server and Dispatch on the same device.

- If **STOREVAR** is used in a way that allows multiple parallel instances, you can use the following syntax for **STOREVAR**:

`data (file name) /M STOREVAR`

The *M* switch disables the **STOREVAR** error detection when VIeC is run in multi-instances mode. Note that when this switch is used, if two jobs are running in parallel, and **STOREVAR** executes with the same file name, an error occurs, or the final content of the file referenced by **STOREVAR** can be incorrect.

## VIEC SERVER AND CLIENT

This is the default configuration that results from a typical single-workstation VIeC installation. The following components are installed on a single workstation:

- Local VIeC Server

- Job Submission Client

- Watched Folders Client

- VIeC Dispatch Server

### Local Job Submission

If you install a VIeC Server and Client on drive C, `user_root` is set automatically to `user_root=C:\Program Files (x86)\Xerox\VIPP\xvtp\users`.

These settings specify that a VIeC Client feeds jobs to a VIeC Server, both on the same workstation, as shown.
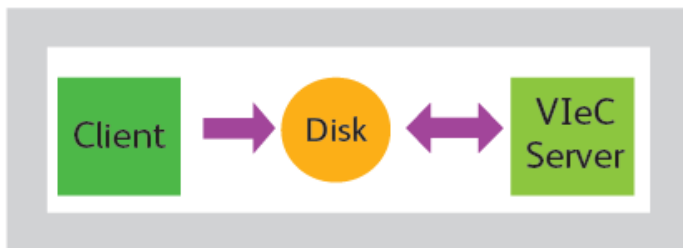


### Remote Job Submission

Optionally, you can specify an existing VIeC Server on the network for use in remote job submission.

When using this option, map a network drive, for example, X, to the remote VIeC Server.

The *remote_user_root* is set automatically to `remote_user_root=X:\ProgramFiles(x86)\Xerox \VIPP\xvtp\users`.

These settings result in the Client feeding jobs either to the local or remote VIeC Server, as shown below:
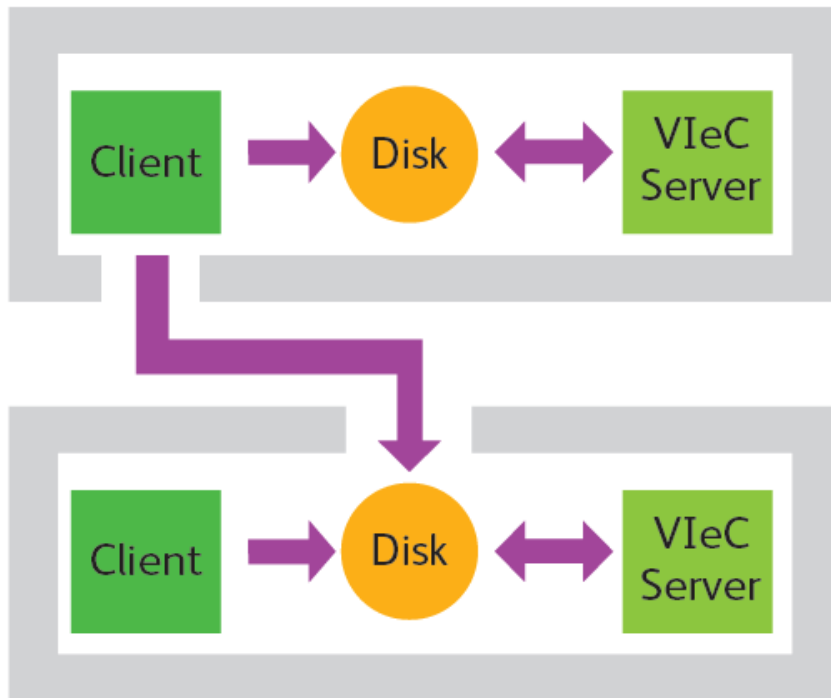
✎ Note: This option is not in any way related to submitting jobs to a cluster configuration. This clarification is made explicitly as there has been some confusion in the past: This option and its associated *remote_user_root* and *remote_server_root* entries were intended for use with a developer license configuration which is no longer available. As such their use is deprecated and is to be ignored, if mentioned in older or other external documentation.

### VIEC CLIENT ONLY

In a Client-only configuration, these components are installed on the workstation:

- Job Submission Client
- Watched Folders Client

In this configuration on drive C, the user_root is automatically set to `user_root=X:\Program Files(x86)\Xerox\VIPP\xvtp\users`, where X is the partition where the VI eCompose software is installed.
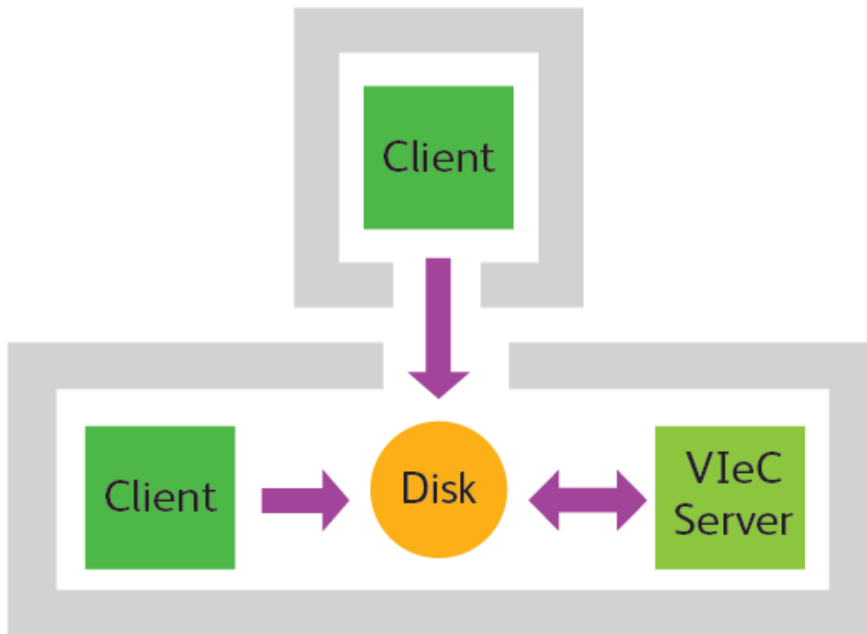
Note also that multiple Clients can be mapped to the same remote VIeC Server.
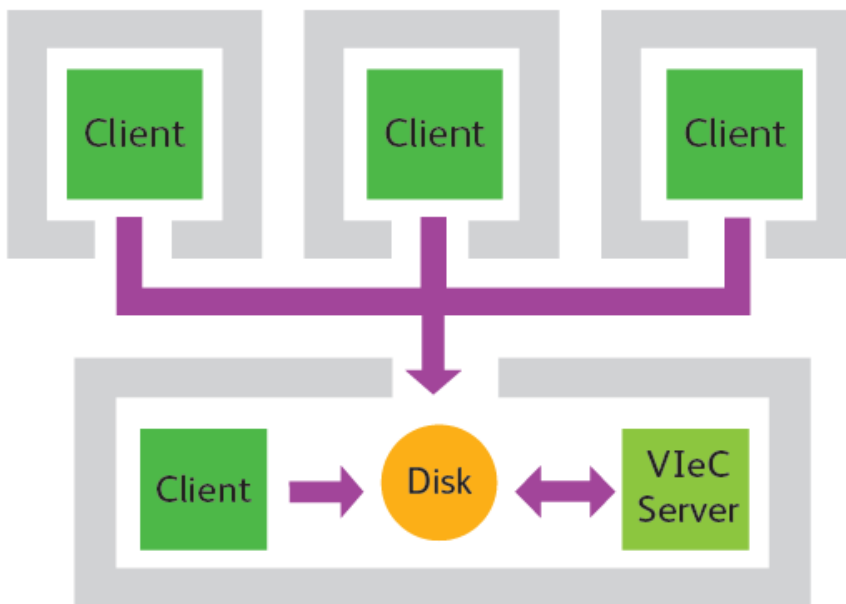


Therefore, a single VIeC Server can be shared as a workgroup resource. However, since the single VIeC Server processes jobs sequentially, this cannot provide satisfactory job turnaround as the size or number of jobs and clients increases.

## VIEC SERVER NODE SERVICING A VIEC CLUSTER

VIeC does not rely on any particular network protocol to submit jobs, consume jobs, or report status of jobs. Instead, VIeC relies on directory structures, disk-based mailboxes, and sentinel files to perform these operations.

VIeC Server administrators control security using file system read/write permissions mechanisms that are native to the operating system where VIeC was installed. Permissions are applied to directories within this structure on a user-by-user or group-by-group basis, as appropriate.

A VIeC Server can be made available for remote use by sharing the server install directory, for example, `D:\xvtp` of the drive on which the VIeC Server is installed. Windows clients can then access the VIeC Server by mapping a network drive on their workstation, for example, X, to the shared location on the server drive and selecting the mapped drive during installation.

**Incoming Jobs Folder**

To provide security for sensitive customer information and to support load-balancing of jobs, the output of jobs submitted to a given VIeC Server, whether on a networked file system or on the local disk, is sent to a pre-designated destination folder within that VIeC Server directory structure.

Administrators of VIeC Servers control security via the file system read/write permissions mechanism. Permissions are applied to directories within this structure on a user-by-user or group-by-group basis, as appropriate.

An example of a VIeC Server installation follows. Note that this example is incomplete for clarity.

```
xvtp
   |- bin
   |     |- xgf
   |           |- decomp
   |           |- demo
   |           |- encoding
   |           |- formlib
   |           |- imglib
   |           |- jdtlib
   |           |- relib
   |           |- src
   |- docs
   |- profiles
   |- users
         |- John_Doe
               |- incoming
               |     |- alt1
               |     |- alt2
               |     |- bulk
               |     |- bulk_satatus
               |     |- default
               |           |- John_Doe_bigjob_lm_2002_10_12_14_08_35
               |                 |- bigjob.cfg
               |                 |- bigjob.pdf
               |                 |- status
               |- outgoing
               |- profiles
```

The incoming jobs directory is created during VIeC Server installation. The incoming jobs directory contains the default incoming folder. Any number of alternate incoming folders can be created under the incoming jobs directory.

During VIeC installation, a shortcut to the incoming jobs directory is placed on the desktop.

As shown in the previous example, John Doe has an incoming directory incoming that contains his default incoming

folder, default. He also has two alternate incoming folders, alt1 and alt2, that he can rename as desired. In addition, he can create additional incoming folders as needed. A shortcut to `X:\Program Files(x86)\Xerox\VIPP\xvtp\users\John_Doe\incoming` has been added to John Doe's desktop for easy access to his incoming folders.

If John wants to submit a simple VIPP® line mode job for distillation, he must specify the destination folder, default in this example, via the Job Submission Client when submitting his job, bigjob.lm. The VIeC Server creates a directory for the new job under the specified destination folder; the directory name is generated using the user name, the job name, and the date and time, for example, `John_Doe_bigjob_lm_2001_10_12_14_08_35`. The generated PDF file is placed in that folder. The `.cfg` file and the status subdirectory are by-products of processing the job and can be ignored. These by-products are used by the Job Submission Client when it displays extended information about the job.

## vipo.ini Configuration Settings

In most situations, the configuration file, vipo.ini, is not to be edited manually; however, you can change some options. If the `vipo.ini` file has been edited, before installation of the new software version, make a copy of the modified file as a backup. After the new version is installed successfully, you can edit the `.ini` file.

On Windows® 10, Windows® 11, Windows Server® 2016, Windows Server® 2019, and Windows Server® 2022, the `.ini` file is in `C:\ProgramData\Xerox\VIPP\VIPO`.

> ✎ Note: On Windows platforms, the `vipo.ini` file is located in the Application Data or Program Data folder, which can be hidden by default, so the folder is invisible to regular users. To make hidden files and folders visible, open the control panel **Folder Options** dialog. In the View tab, select **Show hidden files and folders**.

| VIPO.INI LINES | NOTES |
|---|---|
| relver=xx | Do not change these settings. |
| server_root=C:\Program Files (x86)\Xerox\VIPP\xvtp | |
| user_root=C:\Program Files (x86)\Xerox\VIPP\xvtp\users | You can change the path to use UNC paths. |
| remote_server_root=N /A\xvtp | This entry is deprecated. Ignore this setting. |
| remote_user_root=N/A\xvtp\users | |
| media=USLetter | Do not change this setting. |
| use_ps_dsc_conform_tag = 1 | 1 = true<br>0= false |
| vtp_distill_proc_wait_timeout_hours=24 | The number of hours that VIeC waits before timing out a distilling process. |
| vtp_distill_page_wait_timeout_minutes=60 | The number of minutes that VIeC waits before timing out a distilling page. |
| split_proc_wait_timeout_hours=24 | The number of hours VIeC waits before timing out split processing. |
| split_count_wait_timeout_minutes=60 | The number of minutes that VIeC waits before timing out a split count process. |
| output_filename_strict_checking=true | true = Turn on strict file name checking.<br>false = Turn off strict file name checking. |
| distill_and_split_without_bookmarks=error | error = When bookmarks are not found, an error is returned. |
| split_invalid_bookmark_action=1 | Refer to Using the VIPP® BOOKMARK Command. |
| split_transfer_attributes=true | true = Turn on split transfer attributes.<br>false = Turn off split transfer attributes. |

| VIPO.INI LINES | NOTES |
|---|---|
| jsc_cancel_job_scope=2 | Refer to Function Buttons. |
| pdf_user_password=? | Refer to PDF Security Options. |
| pdf_owner_password=? | Refer to .PDF Security Options. |
| csv_newline_char_seq=(&#x0A;) | Refer to Special Character Sequences in CSV and XML Index Files. |
| csv_dquotes_char_seq=(&quot;) | Refer to Special Character Sequences in CSV and XML Index Files. |
| vtpd_pending_jobs_lo_water=5 | Do not change these settings. |
| vtpd_pending_jobs_hi_water=15 | |
| vtpdispatch_max_preemption_count=10 | |
| v2vwatch_max_timeout_check_minutes=10 | |
| security_options=C:\Program Files (x86)\Xerox\VIPP\xvtp\securityoptions\Default. SecurityOptions | Defines the file in which security options are saved. |
| enable_pdf_security=false | true = Turn on PDF security. <br> false = Turn off PDF security. |
| display_split_warnings=false | true = Turn on the split warning display. <br> false = Turn off the split warning display. |
| v2vwatch_poll_interval_seconds=5 | Sets the polling time in seconds. The default polling time is 5 seconds. |
| v2vwatch_file_size_check_period_seconds=60 | Sets the V2VWatch file size check period time in seconds. The default file size check period is 60 seconds. |
| normalizer_options_file=<default> | |
| enable_mu lti_instance=1 | 1 = On, enable multiple instances. <br> 0= Off, disable multiple instances. |
| vtpserver_number_instances=N | N= The number of instances. <br> Refer to Multiple Instances of VIeC Server and Dispatch as Services. |
| vtpdispatch_number_instances=N | N= The number of instances. <br> Refer to Multiple Instances of VIeC Server and Dispatch as Services. |
| shorten_job_folder_names=0 | To shorten the folder paths to VIeC jobs so that more characters are available for the file name, change the value |

| VIPO.INI LINES | NOTES |
|---|---|
|  | to 1. The character path is 259 characters. |
| enable_distillation_trace_log=0 | 0= Off, disables the trace log.<br><br>1= On, enables the trace log on errors only.<br><br>2= On, enables the trace log on errors and warnings.<br><br>4= On, always creates a trace log.<br><br>The log file is always placed in the same folder as the PDF file. |
| split_match_passwords=error | error = When passwords do not match, an error is returned. |
| split_short_open_password=error | error = When the short open password is incorrect, an error is returned. |
| split_short_permissions_password=error | error = When the short permission password is incorrect, an error is returned. |
| split_short_individual_open_password=error | error = When the short individual password is incorrect, an error is returned. |

## OPTIONAL .INI ENTRIES

**Create a Log File of the Distillation Phase**

To create a log file of the distillation phase:

Log files are not created automatically. To enable the creation of log files one of the following entries must be set in the `vipo.ini` configuration file:

- To create a log file on distillation errors only:

  ```
  enable_distillation_trace_log=1
  ```

- To create a log file on distillation errors or warnings:

  ```
  enable_distillation_trace_log=2
  ```

- To create a log file on every job:

  ```
  enable_distillation_trace_log=4
  ```

# Using VIeC

This chapter contains:

- VIeC does not impose limitations on the length of file names. The Windows operating system has a maximum file name length. A fully qualified file name path has a maximum of 259 characters. The 259 characters include the drive letter, intermediate folder names, backslashes, and a file name. Since all paths begin with $X:\backslash$, where X is a drive letter, a file name path is limited to the remaining 256 characters.

- Because VIeC has job-naming conventions that add a date and time stamp, and optionally, the VIeC user ID to the job folder name, and because of the output directory hierarchy, it is recommended that the length of job file names be kept to 90 or fewer characters. The actual maximum length depends on the job setup, which includes the location of the VIeC users folder, the length of the VIeC user ID, the output destination folder in the incoming directory, and so on.

## VIeC Server and Client

A VIeC Server must be running and accessible by the Job Submission Client in order for VIPP® jobs to be processed via VIeC. Each VIeC Server automatically starts when the workstation on which it is installed boots, and it subsequently runs as a background process.

You can tell a VIeC Server is running by the presence of its icon in the Windows task bar. The icon is located, by default, in the lower-right corner of the desktop. Right-click the VIeC Local Server icon to display these options:

Launch Job Submission Client

Select this option to open the VI eCompose Job Submission Client window.

Launch Watched Folders Client

Select this option to open the VIeC Watched Folders Client window. For more information on the VIeC Watched Folders feature, refer to Using VIeC Watched Folders Client.

Launch Dispatcher

Select this option to open the VI eCompose Dispatch Window. For information on the VIeC Dispatch option, refer to Using VIeC Dispatch.

Restore VI eCompose Local Server

Select this option to maximize the VI eCompose Local Server window, allowing access to more Server options.

Exit

Select this option to exit the Server. If you exit the Server, VIeC jobs will not be processed by that workstation until the Server is restarted. You can restart the Server from the workstation **Start** menu at **Start > Programs > VI eCompose > VI eCompose Server**.

## Using VIeC Local Server

The VI eCompose Local Server window provides access to the File, Job Selection, and Help drop-down menu options.

You can maximize the VIeC Local Server window in one of two ways:

- Double-click its **task bar icon**.

- Right-click its task bar icon, then choose **Restore VI eCompose Local Server** from the menu.

Clicking the **X** button in the upper right corner of the window does not cause the Server to exit; instead it minimizes the Server to its taskbar icon. Jobs are still processed.

To exit the VIeC Local Server, either select **Exit** from the File menu or select **Exit** from the task-bar icon right-click menu.

### FILE MENU OPTIONS

These options are available from the File menu:

- Pause Server: Pauses or unpauses the Server. You must pause the Server before dumping an accounting file.

- Save Logfile: Used when reporting software problems.

- Clear Logfile: Used when diagnosing software problems.

- Dump Accounting File: Saves accounting information to a file, then clears. Refer to Accounting file format for a list of fields included in accounting files.

- Show Billing Meters: Tracks the number of pages distilled, pages split, and PDF files generated on the Server.

  Note: The Billing Meter reading is accurate only when one instance of VIeC is running. When multiple instances are running, billing information is derived from the contents of the accounting logs in the same manner as when running multiple VIeC nodes in a VIeC cluster.

- Exit: Exits the Server; jobs will not be processed.

### Accounting File Format

The accounting file is a field-delimited format file. There is no header on the accounting file. Each record in the accounting file is delimited with commas. These fields are included in each record:

- VIeC Server Name

- User Name

- Input Job Name

- Distill Start Date

- Distill End Date

- Distill Start Time

- Distill End Time

- Distill CPU Time Used, in seconds. This field is not yet available.

- Distill Result, Completed or Failed

- Distill Message, empty if Result is "Completed", otherwise within double quotes

- Number Distill Pages

- Distill PDF Name

- Split Start Date

- Split End Date

- Split Start Time

- Split End Time

- Split CPU Time used, in seconds. This field is not yet available.

- Split Result, "Completed" or "Failed"

- Split Message, empty if result is "Completed", otherwise within double quotes

- Number Split Pages

- Number Split Files

The following is a sample accounting file record:

```
MYSERVER,John_Doe,bigj ob.ps,2001-09-18,2001-09-18,16:35:39,16:36:19,0,

Completed,,1060,bigjob.pdf,2001-09-18,2001-09-18,16:36:20,16:37:05,0,

Completed,,1060,1060
```

When you select Dump Accounting File from the File menu in the VIeC Local Server window, a corresponding XML-formatted Accounting file is created automatically. Both accounting files have the same file name except for the file extension `*.acct` and `*.xml`.

The following is an example of an XML-formatted Accounting file.

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
 <XVTPAF Version="0.1">
  <XVTPDOC>
    <XVTP_Server_Name>XRX-MAVILES-W2K</XVTP_Server_Name>
    <User_Name>maviles</User_Name>
    <Input_Job_Name>credit.vpc</Input_Job_Name>
    <Distill_Start_Date>2002-07-16</Distill_Start_Date>
    <Distill_End_Date>2002-07-16</Distill_End_Date>
    <Distill_Start_Time>01:14:45</Distill_Start_Time>
    <Distill_End_Time>01:14:53</Distill_End_Time>
    <Distill_CPU_Time_Used>0</Distill_CPU_Time_Used>
    <Distill_Result>Completed</Distill_Result>
    <Distill_Message />
    <Number_Distill_Pages>14</Number_Distill_Pages>
    <Distill_PDF_Name>credit.pdf</Distill_PDF_Name>
    <Split_Start_Date>2002-07-16</Split_Start_Date>
    <Split_End_Date>2002-07-16</Split_End_Date>
    <Split_Start_Time>01:14:53</Split_Start_Time>
    <Split_End_Time>01:14:53</Split_End_Time>
    <Split_CPU_Time_Used>0</Split_CPU_Time_Used>
    <Split_Result>Completed</Split_Result>
    <Split_Message />
    <Number_Split_Pages>14</Number_Split_Pages>
    <Number_Split_Files>8</Number_Split_Files>
  </XVTPDOC>
 </XVTPAF>
```

## JOB SELECTION MENU OPTIONS

To determine the order in which the VIeC Server processes jobs, from the Job Selection menu, select one of these options:

- First In First Out (FIFO)

- Shortest Job First (SJF)

- Longest Job First (LJF)

The length of a job is based on the size of the VIPP® file submitted for processing.

## HELP MENU OPTIONS

These options are available from the Help menu:

- VIeC Help: Provides access to the online help facility.

- About: Displays the VIeC Server version information.

## Using the VIeC Job Submission Client

To select and configure a job for submission, use the VIeCompose Job Submission Client window.
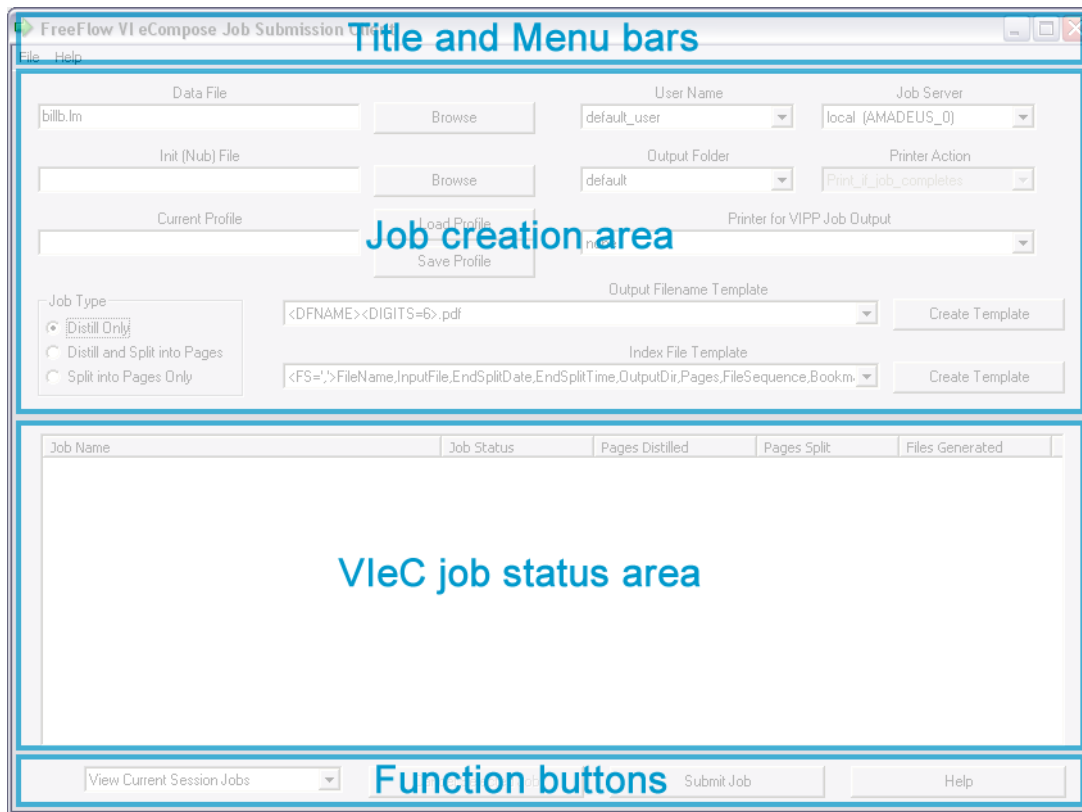


### WINDOW ELEMENTS

The VI eCompose Job Submission Client window consists of these elements:

- Title bar: Displays the product name.

- Menu bar: Provides access to the File and Help menus.

- File: The File menu provides access to the Exit option. Select this option to close the Job Submission Client window. Closing this window while a job is in progress does not cancel the job. The job continues to process to completion unless explicitly canceled.

- Help: The Help menu provides access to the About option, which displays the version information of the VIeC Job Submission Client.

- Job creation area: Contains the fields necessary to select and configure a job for submission. For a description of each of the fields, refer to Job Creation Area Field Descriptions.

- VIeC job status area: Provides the status of each job submitted, including the number of pages distilled and split, and the number of files split.

  When a job submission has been successfully completed, the PDF documents generated can be found and viewed using Adobe Acrobat or Reader. In the Job Status list, select the job folder entry, then double-click to invoke a File Selection dialog. If Adobe Acrobat or Reader is not present or the file cannot be opened, an error message appears.

Note: Adobe Reader is not supplied with VIeC. Obtain Adobe Reader from the Adobe website: www. adobe.com.

**Function buttons**

These buttons appear at the bottom of the VIeC Job Submission Client window:

View Current Session Jobs, View All Pending Jobs, /View All Jobs: Choose a button to view the following:

- Current session jobs that are submitted only within the current session

- All pending jobs from the currently selected user folder that are not yet processed and completed, or whose Cancel requests have not yet been acknowledged by the Server, whether the jobs were submitted before or after the current session began

- All jobs from the currently selected user folder, whether the jobs were submitted before or after the current session began

Cancel Selected Job: Click this button to cancel the job during processing. Before clicking this button, in the VIeC job status area, select the job. Depending on the job processing status at the time of cancellation, the job runs to completion before the cancellation request is initiated by VIeC.

You can select multiple jobs and submit a request to cancel the jobs as a group. By default, only the jobs from your folders and from the folders of the default_user can be canceled. To change the default, use one of these valid settings in the VIeC configuration file `vipo.ini`:

- ## You can cancel jobs only from your folders ##

  `jsc_cancel_job_scope=1`

- ## You can cancel jobs from the folders of default_user ##

  `jsc_cancel_job_scope=2`

- ## You can cancel jobs from any other user folders ##

  `jsc_cancel_job_scope=8`

Note: Job cancellation requests are submitted only by the client. Jobs must be processed by VIeC Server before the jobs can be cancelled. A cancellation request submitted by the client while the server is not running or is in a paused state, does not process until the server executes a VIeC job.

Job Name and Job Status column headers

To sort the list of jobs, left-click Job Name or Job Status.

- **Job Name**: When you left-click Job Name, it sorts the job list by the job name or by the job date-time stamp, in ascending or descending order, one sorting type per mouse click, in the following sequence:

  – Sort by Job Name, in alphabetic ascending order A to Z, from the top

  – Sort by Job Name, in alphabetic descending order Z to A, from the top

  – Sort by the Date-Time stamp, in ascending order with the latest at the bottom

  – Sort by the Date-Time stamp, in descending order with the latest at the top. This option is the default.

- **Job Status**: When you left-click Job Status, it sorts the job list alphabetically, based on the status string in ascending or descending order, one sorting type per mouse left-button click.

Submit Job: To submit the job after completing the appropriate fields in the job creation area, click this button.

Help: To access the VIeC online help facility, click this button.

## JOB CREATION AREA FIELD DESCRIPTIONS

The following fields are in the job creation area of the VIeC Job Submission Client window.

Data File

Enter the name of the data file or browse to locate the VIPP® data file or print file to be submitted. This data file name can be a VIPP® line mode, database mode, native mode, or XML mode print file such as .lm, .dbf, .nm, or .xml, or a VI Project Container .vpc file, or a PDF file with bookmarks.

Init (Nub) File

Enter the name of the init file to be used when processing the job. Specify this file for each raw data file, one without a VIPP® preamble such as STARTLM. For these jobs, the init (nub) file supplies the appropriate VIPP® preamble.

Current Profile

Optional. Load or save a profile using the Load Profile or Save Profile buttons. Profiles are sets of previously used job submission options that save you from re-entering the data. The default folder for loading and saving profiles is the Profiles folder belonging to the current VIeC user, as selected in the Job Submission Client dialog. You can browse any other directories on the VIeC Client system for loading or saving profiles, including the virtual printers profile directory located in the <server_root>/profiles directory. For example: `C:\Program Files (x86)\Xerox \VIPP\xvtp\profiles`.

For more information, refer to Profile Files Defined.

Job Type

Select one of these job types:

| Distill Only | |
|---|---|
| Distill and Split into Pages | Requires that bookmarks are present in the VIPP® job using the BOOKMARK command. If bookmarks are not present in the VIPP® job, the job processes successfully, however, no splitting occurs. |
| Split into Pages Only | PDF files require one or more bookmarks. |

User Name

Enter the user name, which is most often the logon name of the current user. The default_user is used by lpr and is not normally used from within the Job Submission Client.

Job Server

The default option is local. Select the default option, regardless of whether the server is on the same physical machine or out on the network. The remote option is deprecated. Do not select the remote option.

Output Folder

Select the destination folder for the job. The destination folders are named folders on the disks that you do not browse for a path, but choose from named directories that you or an administrator created in a special pre-designated location under control of the VIeC Server.

Printer Action

Conditional action of optional job printing is selected. Select one of the options listed. This option is available only when a Printer for VIPP Job Output has been selected.

| | |
|---|---|
| **print_always** | Sends the job to the selected printer no matter what the outcome of the VIeC job submission. |
| **print_if_job_completes** | Sends the job to the selected printer only if the job completes successfully to VIeC. |
| **Print Master PDF (Raw)** | Sends the Master PDF file to print directly on a specific printer that can handle or process a PDF datastream. |

> 🖉 Note: When Print Master PDF is selected, the PDF file must not be moved from the original destination. When the Print Master PDF option is selected, do not invoke the Dispatch Module to move the Master PDF.

| | |
|---|---|
| **Print Master PDF (Simplex)** | Sends the Master PDF to Adobe Reader, which is used as a proxy to send the PDF to the selected printer for 1-sided or simplex printing. |
| **Print Master PDF (Duplex)** | Sends the Master PDF to Adobe Reader, which is used as a proxy to send the PDF to the selected printer for 2–sided or duplex printing. |

If the selected device does not support the Duplex printing option, the job defaults to Simplex mode.

| | |
|---|---|
| **Print Master PDF (Tumble-Duplex)** | Sends the Master PDF to Adobe Reader, which is used as a proxy to send the PDF to the selected printer for Tumble-Duplex printing. |

If the selected device does not support the Tumble-Duplex option, the job defaults to Simplex mode.

> 🖉 Note: Using the Simplex, Duplex or Tumble-Duplex modes can result in loss of the PDF transparency feature and tends to be slower than sending raw data. When fast throughput is desired, it is recommended that you do not use these features when processing jobs.

🖉 Note:

- VIeC attempts to generate a PDF file and does not submit a job directly to a printer based on the Printer Action setting.
- VIeC sends to the printer the submission file, not the PDF files.

Printer for VIPP Job Output

Specify the printer for printing the job that was submitted to VIeC for processing. If you do not want the job to print, select none.

Only PostScript printers are listed. Ensure that VI Compose is installed on the printer selected for VIPP® job output and ensure that any resources are available for the VIPP® job.

To print a Master PDF, select the appropriate target printer. Ensure that the device is capable of printing PDF files.

Output Filename Template

Select a pre-defined Output Filename template from a list or create an Output Filename template using the Output Filename template wizard. The template you select is applied to the job being processed. The output file name of the PDF files being produced is controlled with the Output Filename template.

This option is available only when the VI eCompose Job Type option is set to Distill and Split into Pages or Split into Pages Only.

For more information on this option, refer to Specifying PDF Output File Names.

Index File Template

Select the default index template or define an Index file template using the Index template wizard. The Index template is applied to the job being processed. Use the Index Template wizard to add additional fields, from the VIPP® BOOKMARK, to the index file generated by VIeC when the job type option is set, to Distill and Split into Pages, or Split into Pages only. VIeC generates a comma-separated file and an XML index file. The index files are used by VIeC Dispatch, or other third-party components to identify individual PDF records and additional data contained in the BOOKMARK for that record. For example, an email address, location ID, or printer address.

This option is available only when the VIeC Job Type option is set to Distill and Split into Pagesor Split into Pages Only.

For more information, refer to Specifying an Index File.

> 🖉 Note: The PDF filename is based on the name of the data submission file. When using the Distill OnlyJob Type, the output PDF filename uses the data submission filename as the output PDF filename. If a single . character is included as part of the data file filename, VIeC considers this a file extension and replaces this information with the .pdf file extension. If the . character is a part of the filename, include an additional . character followed by at least one other valid character. The second period character is treated as the file extension. VIeC replaces the second period with the .pdf file extension, and maintains the first period in the filename.
>
> For example:
>
> - If the submission file name is P31199-SS.234, the resulting PDF filename is P31199-SS.pdf
>
> - If the submission file name is P31199-SS.234.xx, the resulting PDF filename is P31199-SS.234.pdf

Using the other Job Type options allows you to use the Output Filename Template. The value DFNAME, or the option Add Data Filename, results in the same file name as the job submission data file being used as part of the file name template.

**Profile Files Defined**

A profile contains a list of parameters and values that are used to process a job. You can generate a profile from the VI eCompose Job Submission Client screen.

1. In the VI eCompose Job Submission Client screen, select the required options.

> 🖉 Note: Do not include a data file name in the profile. You can use all the other fields.

2. Click **Save Profile**, then assign a profile name.

3. After you save a profile, download it. Click **Load Profile**, then select the required profile. This action populates the options in the VI eCompose Job Submission Client for submitting the job, based on that profile. Profiles are assigned when you use other available submission clients, such as the Watched Folder Client.

A sample profile:

```
[]
InitFile=C:\Users\us970513\Desktop\CoffeeVouchers\coffeevoucher-docucard-A4 Source\Coffee.nub
JobType=distill_only
OutputFolderName=default
OutputFilenameTemplate=<DFNAME><DIGITS=6>.pdf
IndexFileTemplate=<FS=','>FileName,InputFile,EndSplitDate,EndSplitTime,OutputDir,Pages,FileSeq
uence,Bookmark
PrinterName=Xerox ColorQube 9203 PS
PrinterAction=print_if_job_completes
DistillJobOptions=<default>
NotifyOnErrorOrWarning=false
NotificationCommandStr=<empty>
AppendEOFTrailer=false
```

✎ Note: All available job parameter options are listed in the table at the end of this section.

For more profile options, access the VI eCompose Job Submission Client File menu, then select **Additional Profile Options**.

These options are available:

- Include media and finishing options
- Append 'End of File' marker (%%EOF)
- Select Job Options file

Include Media and Finishing Options: Use this option only when the target device for the PDF is the FFPS APPE. The option places media and finishing information into the PDF, so that when processed by the FFPS APPE engine, feeding and finishing are included. The option increases the size of the PDF created, sometimes significantly, depending on the amount of feeding and finishing in the job.

Append End of File Marker (%%EOF): This option inserts a %% EOF end-of-file marker into the data file being processed. The end-of-file marker signifies the end of a print run when processing line-mode data.

Select Job Options File: An Adobe Job Options file specifies to the rendering engine how to process the PDF file. Options such as resolution, color options, and down sampling can affect the size and look of the PDF generated. The option allows you to select any of the default Adobe Job Options files or any Job Option files that you generated using Adobe Acrobat.

Profile Files: A profile requires job parameters for the VIeC job submitted from the lpr command. The table lists the acceptable job parameters.

| JOB PARAMETER | PARAMETER DEFAULT | PARAMETER DESCRIPTION |
|---|---|---|
| AppendEOFTrailer | <empty> false | Acceptable values:<br>• true<br>• false<br><br>When set to true, VIeC inserts the %%EOF marker at the end of the data submission files associated with the profile. An end-of-file marker is sometimes required to terminate a line-mode data file. |
| DistillJobOptions=*filename* | Default | The *filename* is an existing Acrobat Distiller job options file. Refer to Assigning a job options file. The job |

| JOB PARAMETER | PARAMETER DEFAULT | PARAMETER DESCRIPTION |
|---|---|---|
| | | options file must be moved to a location accessible to the PDF authoring engine: `X:\Program Files(x86)\Xerox\VIPP \xvtp\bin\normalizer \settings`.<br><br>When manually entering the name of the Adobe Job Options file, use the fill job options file name. You can enter the file name with or without angled brackets. For example, you can enter the Adobe Job Options file *High_Quality_ USLetter.joboptions* as `High_ Quality_USLetter. joboptions` or as `<High_ Quality_USLetter. joboptions>`. |
| IncludeMediaAndFinishingOptions= true | <empty> false | This option adds feeding and finishing information to the PDF. Use only if the target print device is an FFPS APPE engine. This option increases the size of the generated PDF. |
| IndexFileTemplate=*text* | <FS=':'>FileName:InputFile: <br><br>EndSplitDate:EndSplitTime: <br><br>OutputDir:Pages: <br><br>FileSequence:Bookmark | This option controls header creation and bookmark extraction in the index file that is generated as part of VIeC file-splitting output. |
| InitFile=*text* | No default | (Optional) VIPP® startup file that prepends to the submitted data file. |
| JobType=*text* | distill_only | Acceptable values:<br>• distill_only<br>• distill_and_split into PDF files<br>• split_only into PDF files |
| NotificationCommandStr=*text* | <empty> (optional) | A single command string sent to the Operating System for execution. The command string can include parameters and has a maximum length of 1024 bytes. The command string cannot contain multiple commands separated by a semicolon or other delimiter. |

| JOB PARAMETER | PARAMETER DEFAULT | PARAMETER DESCRIPTION |
|---|---|---|
| | | For more information, refer to System built-in macros.<br><br>✎ Note: If you set the notification command string to the default value `<empty>`, the value disables the command execution, regardless of the value in the NotifyOnErrorOrWarning entry. |
| NotifyOnErrorOrWarning=*text* | false (boolean) | Indicates that a notification command executes when a VIPP® job is processed and completes with warnings or fails due to an error.<br><br>Acceptable values:<br>• false: Disables execution of the Notification command.<br>• true: Executes the Notification command when an error or a warning occurs.<br>• error: Executes the Notification command only when an error occurs.<br>• warning: Executes the Notification command only when a warning occurs.<br>• always: The Notification command is always executed. Upon success, the `JOB_STATUS_MSG` macro contents should be empty. |
| OutputFilenameTemplate=*text* | <DFNAME><DIGITS=6>.pdf | Controls the names created for the PDF output files. |
| OutputFolderName=*text* | Default | Folder location of VIeC job output. |
| Job parameter | Parameter default | Parameter description. |

| JOB PARAMETER | PARAMETER DEFAULT | PARAMETER DESCRIPTION |
|---|---|---|
| PrinterAction=*print-action* | print_if_job_completes | This is a conditional action when you select optional job printing. Select one of these options:<br><br>**print_always**: Sends the job to the selected printer no matter what the outcome of VIeC job submission.<br><br>**print_if_job_completes**: Sends the job to the selected printer only if the job completes successfully by the VIeC server. |
| PrinterName=*text* | None | Specifies the printer for printing the job submitted to VIeC for processing. If you do not want the job to print, select **none**. Ensure that VI Compose is installed on the printer selected for VIPP® job output. Ensure that resources for the VIPP® job are available. Only PostScript printers are listed. |

The default location for loading and saving the profiles is the user Profiles directory. You can select any other directory on the client for loading and saving profiles, including the virtual printers profile directory, in `<server_ root>/profiles`. For example: `C:\Program Files (x86)\Xerox\VIPP\xvtp\profiles`, where `<server_root>` is defined in the `vipo.ini` configuration file.

### SPECIFYING PDF OUTPUT FILE NAMES

You can use a template to specify how PDF file names are formatted during the PDF split operation. To create a template, in the VIeC Job Submission Client window, next to the Output Filename Template field, click **Create Template**. The Create an Output Filename Template window appears.

**Template Syntax**

Enclose the template syntax within matching < >. All other characters, other than characters that are invalid for file names and those described here, are treated as literal characters in the resulting output file name generated by VIeC.

By default, strict file name checking is performed on file name templates. These characters are considered invalid:

\ / : * ? " < > I ' ' ~ ! $ ^ & , ; [ ] { } ( ) % # = + @ tab and blank space

You can modify the default behavior by setting the following value in the VIeC configuration file vipo.ini:

`output_filename_strict_checki ng=false`

By setting the value to false, these invalid characters are considered valid literals in the file name template:

[ ] { } ( ) % # = + @

✎ Note: The square brackets, [ ], must be escaped, preceded by a backslash character \ to be valid literals in the template syntax. For example: `filename_\[OK\].pdf`. Use the other characters without the escape backslash.

| | |
|---|---|
| [Fn] | An optional field number that can appear in any order or number after a DFNAME or BKMARK. For example, [F2] indicates to use field 2 of DFNAME or BKMARK. Valid values for `n` are 0–99. |
| [x..y] | An optional range modifier that can appear in any order or number after a DFNAME or BKMARK. For example, [0..5] indicates to use character 0–5 of DFNAME or BKNAME. Valid values for `x` and `y` are 0–99. |
| <BKMARK[x..y][Fn]> | Uses the bookmark field. The values [x..y] and [Fn] are optional. |
| <DFNAME[x..y][Fn]> | Uses the data file name. The values [x..y] and [Fn] are optional. When producing multiple files, the data file name is the first part of the file name. |
| <DIGITS=n> | The number of digits to use in the output file names when multiple files are produced. If only a single PDF file is produced, no digits are used in the output file name. Valid values for `n` are 1–10. |
| <FS='char'> | A field separator character that is required if [Fn] is used. Valid values for `char` are any printable ASCII character except: [ ] < >. |
| | You can use more than one FS designation. If used within [ ], FS has an effect only within those [ ]. If used outside of the [ ], FS takes effect until the next FS is encountered outside of the [ ]. |
| <GET_INDEX_WITHOUT_SPLIT> | Indicates that the resulting PDF file goes through the split process without actually being split into smaller PDF child files. However, the corresponding CSV/XML index files are generated, which then allows VIeC Dispatch to process the job. |
| | You can submit a Distill Only VIPP® job with the non-split option to generate the job index files needed to process the distilled PDF file with the Dispatch module. |

**Examples**

The following data file name and bookmarks will produce the named files in the table below when the indicated templates are used:

`data file name = mybigjob.lm`

```
pg1 bookmark = account number 01234:xyz
```

```
pg2 bookmark = account number 43210:abc
```

| TEMPLATE SYNTAX | OUTPUT FILE NAMES |
|---|---|
| <DFNAME><DIGITS=3>.pdf | mybigjob001 .pdf mybigjob002.pdf ... |
| <DFNAME[0..4]><DIGITS=5>.pdf | mybig00001.pdf mybig00002.pdf ... |
| <FS=' '><BKMARK[F2][0..4]>_<DIGITS=4>.pdf | 01234_0001 .pdf 43210_0002.pdf ... |

In the last example, the syntax specifies to extract field number 2 (zero based) from the bookmark using a space as the field delimiter, then extract digits 0 through 4 from that result and use that as the first part of the file name. Then append the literal character _, followed by 4 digits that increment with each file produced.

**Template Field Descriptions**

Use the buttons on the Create an Output Filename Template window to build the template. As you click the buttons, the selections appear in the display window. The default template is `<DFNAME><DIGITS=6>.pdf`

You can manually edit the template shown in the display window. Ensure that you follow the template syntax rules.

**Add Data Filename**

Adds <DFNAME> to the displayed template at the current insertion point, if legal, otherwise, <DFNAME> is added at the most logical insertion point.

**Add Bookmark**

Adds <BKMARK> to the displayed template at the current insertion point, if legal, otherwise, <BKMARK> is added at the most logical insertion point.

**Add File Split Count**

Adds <DIGITS=n> to the displayed template at the current insertion point, if legal, otherwise, <DIGITS=n> is added at the most logical insertion point. The n value is specified in the box to the right of the button.

**Add Field Separator**

Adds <FS='char'> to the displayed template at the current insertion point, if legal, otherwise, <FS='char'> is added at the most logical insertion point. The char value is specified in the box to the right of the button.

**Add Field Index**

Adds <Fn> to the displayed template at the current insertion point, if legal, otherwise, <Fn is added at the most logical insertion point. The n value is specified in the box to the right of the button.

**Add Start/End Index**

Adds [x..y] to the displayed template at the current insertion point, if legal, otherwise, [x..y] is added at the most logical insertion point. The x start index and y end index are specified in the boxes to the right of the button.

**Undo Changes**

Undoes the last change. You can undo multiple changes by clicking this button multiple times.

**Redo Changes**

Redoes the last change. You can redo multiple changes by clicking this button multiple times.

**OK**

Saves the changes and closes the template window. The template created appears in the window under the Output Filename Template field in the Job Submission Client window.

**Help**

Provides access to the online help facility.

**Cancel**

Closes the template window without saving any changes.

### SPECIFYING AN INDEX FILE

During the PDF splitting process, two index files are automatically created: an ASCII file formatted in Comma Separated Values (CSV) and a corresponding XML-formatted index file. Both have the same file name, except for the file extension (*.csv and *.xml).

This is a sample of output for an automatically generated XML-formatted index file:

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<XVTPIF Version="0.1">
 <XVTPDOC>
  <FileName>credit000000.pdf</FileName>
  <InputFile>credit.pdf</InputFile>
  <EndSplitDate>2002-07-15</EndSplitDate>
  <EndSplitTime>17:20:13:890</EndSplitTime>

 <OutputDir>C:\xvtp\users\maviles\incoming\default\
     maviles_credit_vpc_2002_07_15_17_20_09</OutputDir>
  <Pages>2</Pages>
  <FileSequence>0</FileSequence>
  <Bookmark>David Kirk</Bookmark>
  </XVTPDOC>
 <XVTPDOC>
  <FileName>credit000001.pdf</FileName>
  <InputFile>credit.pdf</InputFile>
  <EndSplitDate>2002-07-15</EndSplitDate>
  <EndSplitTime>17:20:13:921</EndSplitTime>

 <OutputDir>C:\xvtp\users\maviles\incoming\default\
     maviles_credit_vpc_2002_07_15_17_20_09</OutputDir>
  <Pages>1</Pages>
  <FileSequence>1</FileSequence>
  <Bookmark>Michael Sotelo</Bookmark>
  </XVTPDOC>
 <XVTPDOC>
  <FileName>credit000002.pdf</FileName>
  <InputFile>credit.pdf</InputFile>
  <EndSplitDate>2002-07-15</EndSplitDate>
  <EndSplitTime>17:20:13:968</EndSplitTime>

 <OutputDir>C:\xvtp\users\maviles\incoming\default\
     maviles_credit_vpc_2002_07_15_17_20_09</OutputDir>
  <Pages>3</Pages>
  <FileSequence>2</FileSequence>
  <Bookmark>Carolyn Cliff</Bookmark>
  </XVTPDOC>
 </XVTPIF>
```

You can also use a template to specify the contents of the index file header and index file record. To create a template, click **Create Template** next to the Index File Template field on the VIeC Job Submission Client window. The Create an Index File Template window appears; the default template appears in the display window.

Additional checks are made when creating and supplying an Index File Template to make sure the field names have

the appropriate syntax for XML tag creation. These characters are invalid for template field names:

\ / : * ? " < > | ( ) [ ] { } ' ' ~ ! @ # $ % ^ & + = , ; tab and blank space.

**Template Syntax**

The Create an Index File Template window consists of the following eight fields: seven index file fields and a bookmark field. A defined Field separator bullet character separates each field. For more information, refer to Header Field Separator.

- FileName
- InputFile
- EndSplitDate
- EndSplitTime
- OutputDir
- Pages
- FileSequence
- Bookmark

The first seven fields are names of corresponding fields in each line of the index file generated by VIeC. Only the names of these seven fields can be changed in the Index File Template.

The the eighth field, the bookmark field, can be replaced with as many field names as needed, each separated by the defined field separator. For more information, refer to the Bookmark field bullet and Bookmark Field Separator.

Note: All fields in the index file header and index file record are enclosed in double quotes. This feature cannot be turned off.

| Field separator | A field separator is a single character that separates the index file fields and bookmark fields. The separator is used to generate the field separators in the index file header and in each index file record. You can change the field separator used in the template. The syntax for the field separator is<br><br>`<FS='char'>`<br><br>Where 'char' is any printable ASCII character except: [ ] < >. The field separator in the default template is a comma (,). |
|---|---|
| Bookmark field | You can replace the bookmark field with an unlimited number of field names. Each bookmark field name appears in the index file header.<br><br>Use the following syntax with any bookmark field name in the template to control which portions of the bookmark are extracted and placed into the corresponding fields in each index file entry generated by VIeC. |

> Note: The default template places the entire bookmark into the last field of each index file.

**[x..y]**  Uses a substring of Bookmark from zero-based indices x to y. This is an optional range modifier that can appear in any order or number after the fixed fields in the template. Valid values for x and y are 0–99.

**[Fn]**  Uses field number n of the bookmark according to BFS. This is an optional field number that can appear in any order or number after the fixed fields in the template. Valid values for n are 0–99.

**<BFS='c'>**  A bookmark field separator (BFS) that must be specified if [Fn] is used. Valid values for c are any printable ASCII character except: [ ] < >.

You can use more than one BFS designation. If used within [ ], it only has an effect within those [ ]. If used outside of the [ ], c takes effect until the next BFS is encountered outside of the [ ].

**Examples**

The following bookmark contents will produce the index file headers and index file records below when the indicated templates are used:

`bookmark contents = John Doe 123-45-456`

Example 1:

| Template | <FS=':'>Fld1:Fld2:Fld3:Fld4:Fld5:Fld6:Fld7:BookFld1\|<BFS=' '>[F1]:BookFld 2\|[0. .2] :BookFld3\| [F2] [0..2] |
|---|---|
| Index file header produced | Fld1:Fld2:Fld3:Fld4:Fld5:Fld6:Fld7:BookFld1:BookFld2:BookFld3 |
| Index file record produced | outfi1e_001.pdf:infile:10-2-01:"23:56:34":"c:\user\...":5:1- 50:Doe:Joh:123 |

Example 2:

| Template | <FS=':'>FldA:FldB:FldC:FldD:FldE:FldF:FldG:Bookfield1\|<BFS=' '>[F1]: Bookfield2\|[F2][<BFS='-'>[F2]] |
|---|---|
| Index file header produced | FldA:FldB:FldC:FldD:FldE:FldF:FlgG:Bookfield1:Bookfield2 |
| Index file record produced | outfile_003.pdf:infile:10-2-01:"13:06:14":"c:\user\...":7:3-80:Doe:456 |

Example 3:

| Template | <FS=','>Fld1,Fld2,Fld3,Fld4,Fld5,Fld6,Fld7,Fi rstNaml <BFS=' '>[F0],LastNam l[F1],Numl[F2][<BFS='-'>[F0]] |
|---|---|
| Index file header produced | Fld1,Fld2,Fld3,Fld4,Fld5,Fld6,Fld7,FirstNam,LastNam, Num |
| Index file record produced | file_xyz.pdf,infile,10-7-01,14:29:14,c:\user\...,4,10-100, John,Doe,123 |

## TEMPLATE FIELD DESCRIPTIONS

To build the template, use the buttons in the Create an Index File Template window. The selections appear in the display window. The default template is shown:

```
<FS=':'>FileName:InputFile:EndSplitDate:EndSplitTime:OutputDir:
Pages:FileSequence:Bookmark
```

Note: You can manually edit the template shown in the display window. Ensure that you follow the template syntax rules.

- **Header Field Separator**: Adds the index field separator to the displayed template at the current insertion point, if legal, otherwise, the separator is added at the appropriate insertion point. In the box to the right of the button, specify the field separator.

- **Bookmark Field Separator**: Adds the bookmark field separator to the displayed template at the current insertion point, if legal, otherwise, the separator is added at the appropriate insertion point. In the box to the right of the button, specify the field separator.

- **Add Field Index**: Adds <Fn> to the displayed template at the current insertion point, if legal, the separator is added at the most logical insertion point. Fn is specified in the box to the right of the button.

- **Add Start/End Index**: Adds [x..y] to the displayed template at the current insertion point, if legal, otherwise, [x.. y] is added at the most logical insertion point. The x start index and y end index are specified in the boxes to the right of the button.

- **Undo Changes**: Undoes the last change. You can undo multiple changes by clicking this button multiple times.

- **Redo Changes**: Redoes the last change. You can redo multiple changes by clicking this button multiple times.

- **OK**: Saves the changes and closes the template window. The template created appears in the window under the Index File Template field in the Job Submission Client window.

- **Help**: Provides access to the online help facility.

- **Cancel**: Closes the template window without saving any changes.

## Using VIeC Watched Folders Client

The VIeC Watched Folders Client (WFC) allows you to specify one or more directories where VIPP® jobs can be placed for automatic submission to VIeC. A VIeC remote or local user and an optional profile are associated with each watched folder, which controls the processing of all jobs submitted from that folder. Submitted jobs can be monitored using the Job Submission Client in the same manner as for lpr-submitted jobs. All VIPP® jobs found in each watched folder are checked at each watch interval to determine file readiness.

> Note: Due to the overhead required to monitor Watched Folders, it is recommended that no more than six Watched Folders be defined for each system. Although there is no hard limit on the number of Watched Folders you can define, the hard limit for the Watched Folder names is 3072 bytes. Assuming that the average Watched Folder name `watchedFolderPath,userName (scope),profileName.pfl` can contain 200 bytes, this would make a practical limitation of 15 Watched Folders.

You can start the Watched Folders Client from the VIeC program folder or from the right-click menu of the **VIeC Local Server** icon. The Client runs as a background process. A VIeC Server must be running for the watched folders to process.

You can tell the Watched Folders Client is running by the presence of the icon in the Windows task bar. To display the Watched Folders Client options, right-click the **VIeC Watched Folders Client** icon. The options appear:

- **Restore VI eCompose Watched Folders Client**: Maximizes the VIeC Watched Folders Client window.

- **Exit**: Exits the Watched Folders Client. If you exit the client, folders are not watched until the client is restarted.

The VI eCompose Watched Folders Client window is used to control and monitor watched folders.

> Note: When the client is invoked, the watch process is inactive.

### WINDOW ELEMENTS

The VIeC Watched Folders Client window consists of these elements:

| Title bar | Displays the product name VI eCompose Watched Folders Client. |
|---|---|
| Menu bar | Provides access to the File and Help menus. These options are available from the File menu:<br><br>**Hide Watched Folders Client** Returns the VIeC Watched Folders Client window to a task-bar icon.<br><br>**Input File Disposition** Controls what happens to watched jobs after they are submitted to VIeC. There are two submenu choices:<br><br>**Delete Input File**: Deletes the job from the watched folder. This option is the default.<br><br>**Move Input File to Saved Jobs Folder**: Moves the job to the Saved Jobs folder contained under each watched folder.<br><br>**Exit** Exits the Watched Folders Client. Folders are no longer watched. |

| | Help The Help menu provides access to the About option, which displays the version of VIeC Watched Folders Client. |
|---|---|
| Watched folders list | Lists the folders currently defined for watching. |
| Watched folder details area | Shows the details for the selected watched folder in the watched folders lis: the folder name, VIeC user associated with the folder, and the profile. If a profile is specified, the contents of the file appear in the Profile Contents box. The Browse button allows you to browse for the folder you want to add to the watched list. |
| Status area | Displays the current watch interval, which can be changed using the Change Interval button, and the current method used to check file readiness. The status area indicates if folders are being watched currently. |

**Function Buttons**

These buttons appear on the right side of the VIeC Watched Folders window:

**Resume/Pause**

Resume starts or resumes watched folder processing. Pause stops the Watched Folders Client from watching the folder and no further jobs are queued. Up to 120 jobs that have been submitted prior to selecting the Pause button continue to process until the jobs are completed.

**Change Interval**

Changes the current watch interval. When you click the Change Interval button, a Change Watched Folders Interval window appears that allows you to change the interval in seconds. The status area in the VIeC Watched Folders window shows the watch interval.

**File Ready Check**

Used to select the method to determine when a file is ready for job submission, preventing processing and submitting files that are not yet ready for job submission. This selection is indicated in the Status area of the window. For more information, refer to File Ready Check.

**Add Folder**

Used to expand the dialog where the watch folders are managed. In the expanded dialog, you can add, delete, and save the changes in the watched folder.

**Add**

Allows you to add a folder to the watched list.

**Delete**

Allows you to delete the selected folder from the watched list.

**Save**

Saves the current changes for the selected folder.

**Close**

Closes the dialog that is expanded when you click the **Add Folder** button.

**ADDING WATCHED FOLDERS**

To add a new watched folder:

1. Click **Browse**, then select the new folder. The folder appears in the Watched Folder box in the watched folder details area.

2. From the VIeC User selection box in the watched folder details area, select the user. The selected user is used for all the jobs submitted from that new watched folder.

3. From the Profile File selection box for the jobs to submit to the watched folder, select the profile.

4. To add the new watched folder to the current list of watched folders, click **Add Watch**.

**FILE READY CHECK**

The file ready check feature allows you to choose the method you want VIeC to use to determine if a file is ready for use in a VIeC application. Using this feature prevents problems that can occur when the WFC encounters a file that is not ready for submission. For example, a large file that is still being transferred or copied from a remote location or files being built on the fly by an automated process.

The File Ready Check button invokes the Method to Check for File Readiness dialog.

The dialog offers two choices to check for file readiness before a file is picked up for job submission:

**Check for Existence of Flag File**

Used to signal the WFC directly when a file is ready for job submission by creating a flag file. The flag file is an empty, 0-byte file that you create. In addition to having the same name as the job file, the flag file must have the additional extension: ._File_Ready_For_Submission_.

The flag file for myVIProject.vpc, becomes: `myVIProject.vpc._File_Ready_For_Submission_`

Copy the flag file to the same watched folder as the job file. The flag file uses the naming convention described. After the job file has been successfully transferred, copy the flag file to the watched folder only. If, during the polling intervals, the flag file is not found, it is assumed that the file transfer has not completed. It is only when the flag file is found that the WFC makes the job file available for processing.

**Check for File Size Changes**

The Check for File Size Changes is the default setting in which the WFC checks and monitors changes in the file size at every watch interval. If the size of the file for submittal does not change within the specified period of time, the file is considered ready for job submission and is processed at that time, even if a flag file does not exist. The range of time is 4–1800 seconds. The default threshold is 60 seconds.

When using the Check for File Size Changes option, if large files are transferred or copied from a remote location to the watched folder, and the time it takes for the file transfer is longer than the current watch interval, the file size changes during the transfer are monitored and the file is not picked up for job submission until file size growth is not detected within the specified time. Choose a period of time for the watch interval, taking this information into consideration:

- The file size changes to be detected by the operating system

- The speed of file transfer or copy

When the Check for File Size option is active, files may no longer be picked up and submitted for processing when found at every watch interval. The effective submission rate is based on the file size check threshold as well as the watch interval. Even for small files, job submission takes place after the threshold is reached, assuming the watch interval is smaller than the threshold time.

For example, a small threshold of 5 seconds can be used for small files. For large files, change the threshold accordingly.

## COMMAND LINE OPTIONS FOR VIEC WATCHED FOLDERS CLIENT

VIeC Watched Folders Client has two command parameters:

`v2vwatch autostart noshow`

These parameters are optional and can appear in any order or case on the command line.

| autostart | VIeC Watched Folders Client starts watching folders immediately. Press the Resume button on the Watched Folders Client dialog. |
|---|---|
| noshow | VIeC Watched Folders Client starts in iconized or minimized size. |

Both of these options facilitate the automatic command-line start of the VIeC Watched Folders Client.

Note: Depending on the system environment, the full path name of the VIeC Watched Folders Client executable needs to be specified. For example: `C:\Program Files(x86)\Xerox\VIPP\xvtp\bin \v2vwatch.exe` param1 param2.

## Bulk Processing

✎ Note: The capability to support bulk processing has been retained to support backward compatibility with older VIeC configurations. VIeC Dispatch provides an intelligent way to control the PDF files generated by VIeC. If you are considering using bulk processing capabilities, consider using VIeC Dispatch instead.

Bulk processing allows a VIeC user to direct the PDF output of multiple jobs to a single bulk directory, typically to facilitate batch or other automated post-processing of VIeC-generated output.

The VIeC system administrator must ensure that the system is upgraded to a VIeC software version that supports bulk processing. If you do not upgrade the system, jobs submitted to the bulk folder are processed in the same manner as non-bulk jobs and the results are generated in individual subfolders on a per-job basis.

To enable bulk processing, if not automatically created during installation, each user is required to create two additional directories, `\incoming\bulk` and `\incoming\bulk_status`, as shown:

```
VIeC
   |-bin
   |-docs
   |-profiles
   |-users
        |-default_user
            |-incoming
                 |-alt1
                 |-alt2
                 |-bulk
                 |-bulk_status
                 |-default
```

Jobs sent to a bulk destination folder have the PDF output sent to `\user\incoming\bulk`, while the status folders and files for those jobs accumulate in `\user\incoming\bulk_status`. All PDFs generated using Distill Only, Distill and Split, and Split Only options, and the index file for each submitted job, are placed in `\user \incoming\bulk`.

✎ Note: Ensure that the job and file-naming conventions and split templates do not cause output file name clashes that result in overwritten files.

## Housekeeping

During job processing, VI eCompose creates job folders and files that are left behind after each job is completed. It is highly recommended to set up a separate cleanup process that regularly removes or archives old job folders. At least once a month, or more frequently when running a large number of jobs every day, run the cleanup process. Existing old job folders accumulate in the VI eCompose user directories and can significantly slow down the search task that looks for jobs to process. It is recommended that you clean up the temporary folders used by the system for the user accounts running in the VI eCompose Server.

On Windows® 10, Windows® 11, Windows Server® 2016, Windows Server® 2019, and Windows Server® 2022, a user temporary folder is located on the C drive:

`C:\Users\[USERNAME]\AppData\Local\Temp\`

Where *[USERNAME]* is the User Account name for the VI eCompose server.

# Using BOOKMARK, PDF Features, and Project Containers

This chapter contains:

## Using the VIPP® BOOKMARK Command

VIPP® BOOKMARK can be used in any VIPP® mode to generate a bookmark on the PDF document. The syntax is:

```
(bookmark contents) BOOKMARK
```

Where `bookmark contents` is the text that will appear on the PDF bookmark generated by VIeC. Variables are allowed.

The following example will create a PDF bookmark containing `John Doe 687863456` which is John Doe's name and social security number, on the specified page.

### DBM file:

```
600 300 MOVETO
($$FNAME. $$NAME. $$SS#.) VSUB BOOKMARK
PAGEBRK
```

DBF file:

```
%!
XGF
FNAME:NAME:SS#
John:Doe:687863456
```

### EXTENDED BOOKMARKS

An extended bookmark is a transparent PDF note, which is usually visible from the comment section of the completed PDF only. Extended bookmarks are used only with the dispatch function of VIeC.

Extended bookmarks can contain information up to 64 K. Regular bookmarks contain up to 256 characters. The contents of the extended bookmark is treated as follows:

- Appended to the contents of the regular bookmark

- Linked to a regular bookmark

- Created after the regular bookmark

Several extended bookmarks can be created in a sub-document between two consecutive regular bookmarks.

This syntax is used to create an extended bookmark:

```
(bookmark contents) [ /EX optl opt2 ] BOOKMARK
```

Where:

| | |
|---|---|
| **/EX** | Tells BOOKMARK to create an extended bookmark rather than a regular bookmark. |
| **opt1** | An integer that tells the VIeC splitter how to handle the child PDF file:<br>• 0 Do not produce the child PDF file<br>• 1 Produce the child PDF file but do not include bookmark in it.<br>• 2 Produce the child PDF file and include a bookmark in it. This is the default. All extended |

| | |
|---|---|
| | bookmarks within a sub-document require the same opt1 value. |
| **opt2** | An integer that tells the VIeC splitter how to handle the extended bookmark:<br><br>• 0 Do nothing with the extended bookmark.<br><br>• 1 Merge the extended bookmark with the associated regular bookmark for index file processing, but do not include the bookmark in the child PDF file.<br><br>• 2 Merge the extended bookmark with the associated regular bookmark for index file processing and include the bookmark in the child PDF file. |
| **opt2=0** | Can be used to insert comments or instructions in the main PDF file present only in this file. |

The combination opt1=0 and opt2=2 is irrelevant. The command acts as opt2=1.

This example creates an extended bookmark that instructs the VIeC splitter to produce a child PDF file that does not contain the extended bookmark. However, this extended bookmark is included in the VIeC index file and, with appropriate VIeC Dispatch and mail server settings, triggers an email to John Smith with the child PDF file attached.

```
(John Smith@isp.com:April invoice:Dear John,\nAttached you will find your
invoice for April.\nBest regards,\nPaul Martin) [ /EX 2 1 ] BOOKMARK
```

## Using VIPP® PDF Interactive Features

A complete set of VIPP® PDF Interactive Features (PIF), available in VIPP® 4.0 and later, allow you to create interactive elements when the VIPP® job is rendered into a PDF document. For an overview of VIPP® PIF and details about each related command, refer to the *VIPP® Language Reference Manual.*

## PDF/X Support

VIeC includes partial support for PDF/X. Only the most important PDF/X-3 and PDF/X-1a constructs are recognized and passed from the master PDF to the child PDF. To generate valid PDF/X child documents via the splitting process from a VALID PDF/X master, you must set the following entry in vipo.ini (located in the Windows folder) and restart VIeC:

```
split_transfer_attributes=true
```

Attributes for the child documents will be set to match the master's attributes.

## Using VI Project Containers

VI Project Containers VPC or .vpc, are a convenient way to archive, exchange, and deploy VIPP® applications. VPCs are created with VI Design Pro software.

When creating an application with VI Design Pro, you work typically with a sample print file containing a limited number of records. The sample print file is shown as a local resource in the notebook display and is the file with the blue gear icon. By definition, this is sample data and is not intended for use in a production run.

VIeC software is intended for use in a production environment because it knows about VI Project Containers and does not use the VPC sample data if a production submission file is also in the VPC.

Note: A VPC sent to the VIeC Server requires a production submission file, however, a VPC submitted to the VIeC Server that contains only a sample data file is not rejected and is submitted for processing.

To add a production submission file to the VI Project from within VI Design Pro, follow these steps:

1. From the notebook display, right-click the **Local folder** icon. A menu appears.

2. To add the production submission file to the project, from the menu, select **Add Resource To Project**. Submission files require an extension of .lm .nm, or .dbf. The submission file has a small number 1 in the file icon, indicating that it is submission file 1.

Note: VI Design Pro allows you to add multiple submission files to the project, but the VIeC Server processes submission file 1 only, and all other files are ignored.

3. Export the project as a VPC from VI Design Pro. On the VI eCompose tool bar, click **Export VI Project Container**.

# Using lpr to Submit VIeC Jobs

This chapter contains:

Using lpr to submit VIeC jobs provides the lpr command syntax for the Windows OS.

## Characters in job files

Job files that are submitted through an lpr command cannot contain blank spaces, pound signs (#), equal signs (=), or colons (:) in the file names.

## LPR requirements

- Ensure that both the TCP/IP Print Server and Print Spooler services are installed and activated on the system. On Windows Server® 2016, Windows Server® 2019, Windows Server® 2022 and Windows® 10, Windows® 11 systems, the TCP/IP Print Server service is a named LPD Service and that is not enabled by default. Install the LPD Service, enable the LPD Service, then install the VI eCompose LPR Printing submission feature.

- When submitting jobs to VIeC through lpr commands, the VIeC Server console shows the job names with extra numeric tags, or IDs, appended to the file extension. Because all lpr jobs are sent to a common spool area, the extra tags are needed to avoid file name collisions and guarantee unique name identifiers. The unique tags generated by the system are appended to the original files before file submission and processing by the VIeC Server. The resulting files do not have the extra numeric tags.

## Lpr Client

Instructions for adding one or more lpr clients are provided in lpr job submission installed on a VIeC Server in the installation section of this document.

## Lpr Command Syntax for the Windows OS

`lpr –S servername –P printername printfilename`

Windows does not accept the -T option or parameters on the lpr command line. Any lpr jobs use the defaults as described in Parameter defaults.

For submission of .pdf, .vpc, or other non-text jobs, ensure that you use the -o l option on the lpr command. The lpr command is the last parameter before the job file name. For example:

`lpr –S HostName –S XVTPrinterName –o l BinaryJobFile`

For more information on lpr command parameters, on the command line, type `lpr`.

Note: The `–o l` option can be used for all data file submissions, as it is preferable for the job data stream to be treated as raw bytes when the print spooler takes the file from lpr. For example:

`lpr –S ServerIP –P XVTPrinter –o l JobName`

Where:

| | |
|---|---|
| -o l | indicates that the submission file is treated as raw data and passed to VIeC unfiltered. If not specified, some submission files are lost and no files are submitted to VIeC. |

## Parameter Definitions

**printfilename**

The printfilename is the VIPP® print file that is submitted to VIeC. The print file can be a VPC file or just the data submission file of a job that has been previously deployed on the system where the VIeC server is running.

**printername**

The printername is an lpr-to-VIeC printer defined on a VIeC Client on Windows running an lpd server.

Note: You can create the initial printer name in two ways:

- From the VIeC program folder, use the Install VI eCompose LPR Option.

- When you install VIeC, use the lpr option.

**servername**

The servername is an IP address or hostname of a VIeC Client running lpd. The host name is defined in the file `winnt/system32/drivers/etc/hosts.`

## Parameter Defaults

When using the lpr command for job submissions to VI eCompose, the user destination is determined in this order:

1. If the user that is logged onto the system has a folder in the VIeC users directory, the folder becomes the default folder for all lpr submissions sent with the user account. For example, if John Smith is the user:
   `../xvtp/users/JohnSmith`

2. The default_user folder `../xvtp/users/default_user` becomes the default for all lpr submissions if the user that is logged onto the system does not have a folder inr the VIeC users directory.

The default profile used for lpr submissions is determined in this order:

1. If the lpr_default.pfl profile exists under the current default_user folder (e.g. `../xvtp/users/ default_user/profiles/*.pfl` or `../xvtp/users/JohnSmith/profiles/*.pfl`) this will be used by the lpr submission.

2. If the VIeC lpr printer name is XVTPrinter the XVTPrinter.pfl profile will be used if it exists under the generic profiles directory (`../xvtp/profiles/XVTPrinter.pfl`).

3. If none of the above apply, internal defaults will be used.

Given the above, if you always want to use the default_user folder as the default destination for all lpr submissions you must make sure that the user who is currently logged-on to the system does not have a folder under the VIeC users directory.

## Example Lpr Job Submissions

This section provides three examples of jobs submitted via lpr.

1. This job prints the xyz.lm file using lpr_default.pfl of the default_user if a profile exists; otherwise it uses XVTPrinter.pfl from the generic profile directory:

   ```
   lpr –S HostName –P XVTPrinter –o l xyz.lm
   ```

2. This job prints the xyz.lm file using lpr_default.pfl of the user johndoe if a profile exists; otherwise it uses XVTPrinter3.pfl from the generic profile directory:

   ```
   lpr –S HostName –P XVTPrinter3 –o l xyz.lm
   ```

# VIPP® Job to PDF File Conversion

This chapter contains:

## PDF Security Options

To set PDF security options while originating PDF documents, from the Job Submission Client, use the VI eCompose Security Options dialog. To open the dialog, when the Job Submission Client is open, either press **Ctrl-S** or go to **File > Security Options**.

🖉 Note: If the VI eCompose Server is not installed on the system, the Security Options dialog is not accessible from the Job Submission Client.

### VIEC SECURITY OPTIONS DIALOG

Use the Security Options dialog to enable or disable PDF Document Security, verify current security options, modify settings, and enter passwords. When you click **Save** or **Save As Options**, the interface automatically updates the SecurityOptions file with the new settings.

There are two levels of security applicable to PDF documents:

| | |
|---|---|
| Open | Locks the PDF and restricts file open and read access. If the Open password is set, the PDF document cannot be opened unless the password is supplied. |
| Permissions | Locks all of the PDF document security settings in the file. If the Permissions password is set, none of the security settings of the PDF document can be changed unless the password is supplied. |

**Master PDF**

If you want the master or parent PDF to be secured during distillation, or access a secured master PDF for splitting, select **Enable Password Security**, then provide passwords.

If a valid Open password is provided, VIeC uses this password for two purposes:

- To secure the master PDF during distillation with file open and read access restrictions.

- To access a secured master PDF for splitting purposes.

VIeC does not transfer this password to any PDF file generated because of the split process. This means that, if the original PDF file was produced with an Open password, this Open password is not set for any of the split files generated. If you want to secure split files with individual Open passwords, refer to Split PDFs and Examples: Personalized Open Passwords For Split PDFs.

If a valid Permissions password is provided, VIeC uses this password and security options to secure the master PDF during distillation. If the master PDF is found to be secured during the splitting process, VIeC automatically transfers its permissions settings to each of the split files generated. As a result, all child PDF documents inherit all the security settings of the parent PDF file and have the Permissions password set. Security settings passed on to child PDF documents do not include the Open password from the parent PDF. Those security settings are locked, and regular users cannot change the settings unless they know the password.

If you select **Enable Password Security**, Open and Permissions passwords cannot match, both be empty, or contain less than the minimum required 6 characters.

✎ Note: If you have a master PDF that was secured with a shorter Open password before VIeC 11.0, and you want to split the PDF in VIeC 11.0 and later, the following workaround allows the splitter to obtain the shorter Open password, so it can access the master PDF for splitting:

- Select **Enable Password Security**

- Enter the shorter Open password

- Deselect **Enable Password Security**

- Select **OK**

If you select **Show**, the corresponding password shows in clear text. Otherwise, the password is obscured with asterisks.

The Security Options section remains disabled until the Permissions password has at least 6 characters. This section contains security settings that can be enabled or disabled.

✎ Note:

- Certain security options can override other security options. For example, if you select **Comments** and deselect **FillAndSign**, the result is a secured PDF that has both Comments and Fill In And Sign set.

- There is no way to set page extraction.

**Split PDFs**

If you want split PDFs to have their own individual Open passwords, select **Enable Open Passwords**, which enables the Job Index Field Nameand its menu.

The Job Index Field Name menu contains bookmark field names that were specified in the Index File Template, after the FileSequence field. Choose the bookmark field name that contains the Open password values for split PDFs.

For an example of how this feature can be used to set up a job to split PDFs with personalized Open passwords, refer to Examples: Personalized Open Passwords For Split PDFs.

The VIeC Splitter verifies these individual Open password values during the splitting process. If an Open password matches the Permissions password, or if an Open password is not empty but is less than the minimum required 4 characters, an error or warning, as specified by the INI entry, appears and the Splitter can stop. For more detail, refer to VIeC Configuration.

### VIEC CONFIGURATION

The VIeC configuration file has entries that specify the SecurityOptions file, whether to produce errors or warnings if certain requirements are not met, and enable or disable PDF security, if needed. For Windows® 10, Windows® 11, Windows Server® 2016, Windows Server® 2019, and Windows Server® 2022, the file location is `C:\ProgramData\Xerox\VIPP\VIPO\vipo.ini`:

```
security_options=C:\xvtp\securityoptions\Default.SecurityOptions
split_short_individual_open_password=warning
split_match_passwords=warning
enable_pdf_security=true
```

There are requirements, such as a minimum of four characters for the individual password, if enabled and not empty, and the individual Open password cannot match the Permissions password. When a requirement is not met, a warning appears as default, and the VIeC Splitter notes that a warning occurred. The VIeC Splitter continues, but it is possible that the split PDFs are not fully secured. If you want the VIeC Splitter to stop immediately instead, change the appropriate INI entry to `error`instead.

If split_short_individual_open_password is set to warning and an individual Open password is not empty but is less than four characters, a warning appears. The corresponding split PDF is still secured as usual, and the user can open it with the shorter Open password.

If split_match_passwords is set to warning, and the Permissions password matches an individual Open password in the data, a warning appears. If the master PDF was secured, permissions and security options are transferred to the split PDF, but an Open password is not set for the split PDFs. If the master PDF was not secured, an Open password is set for the split PDFs, but the Permissions password and security options are not set.

In other words, from a user perspective: Your bank produces 3,000 bank statements. Customers can use their personalized password to open their own statement. Assume that your bank has a master statement that is secured, and you do not want customers to change any of the statement settings. If a customer has a personalized password that matches the Permissions password of the bank, anyone can open the resulting split PDF for this customer. This is possible because the Open password is not set, and the customer can enter successfully their Open password as the Permissions password and manipulate the statement. If this outcome is unacceptable, to stop the Splitter and therefore the production of a compromised split PDF, in the vipo.ini file, set split_match_passwords= error.

The Security Options GUI includes Open passwords for Split PDFs. There is now a version 2 of the SecurityOptions file, which includes EnableMasterPDFSecurity=true/false, so VIeC does not rely on enable_pdf_security in the INI file anymore. The only exception is when VIeC deals with a version 1 of the SecurityOptions file that has not been upgraded automatically with a Save or Save As action. In that case, VIeC checks for enable_pdf_security in the INI file.

> 🖉 Note: On the Windows platform, the vipo.ini file is in a folder, Application Data or ProgramData, which can be hidden by default, so it is invisible to regular users. To make hidden files and folders visible, open the **Control Panel Folder Options** dialog. In the View tab, select the option to show hidden files and folders.

### PERSONALIZED OPEN PASSWORDS FOR SPLIT PDFS

For information about personalized open passwords for split PDFs, refer to the following examples.

**Example 1**

The following is an example of how you can set up jobs to produce split PDFs with personalized Open passwords, which involves four steps:

1. Add a password field and values to a database file (*.dbf). Note the sample Password field name and values in bold text.

```
(,) SETDBSEP
(bookmark1.dbm) STARTDBM
Fname,Lname,Emailfield,Combo1,AcctNo,Password,Local,Long
david,Kirk,david.kirk@xerox.com,A&E,12345,david,12,87
anne,brown,anne.Brown@xerox.com,C1,29837,anne,15,65
thomas,smart,Thomas.Smart@xerox.com,M1,98273,thomas,21,34
andrew,gibson,Andy.Gibson@xerox.com,C1,33876,andrew,10,23
mary,williamson,mary@xerox.com,M1,38762,mary,13,85
Jenny,Morgan,jenmorgan@xerox.com,M1,11123,jenny,22,45
%%EOF
```

2. Create a bookmark in the database master (*.dbm) that includes password field name. Note the sample Password field name in bold text.

   ```
   ($$Fname. $$Lname. $$Password. $$AcctNo.) VSUB [2 RED 3] BOOKMARK
   ```

   As a result of this example BOOKMARK command, a master PDF created from distillation contains bookmarks that include four fields. Note that the sample password values from step 1 are included in the third field of the bookmarks.

3. Before you split the master PDF, go to the Job Submission Client. Next to Index File Template, click **Create Template**. As we saw from the master PDF bookmarks in step 2, there are four bookmark fields, first name, last name, password, and account number. Set up the syntax and use your preferred bookmark field names after the first standard seven fields, FileName through FileSequence. In this example, the bookmark field name for password values is now splitpdfpwd.

4. Open the Security Options GUI. In the Split PDFs section, select **Enable Open Passwords**. This option enables the Job Index Field Name and its menu, which includes the bookmark field names that you specified in the Index File Template, after the FileSequence field, in step 3. Select the bookmark field name that contains the password values, which is splitpdfpwd in this example.

Now future split PDFs are secured with individual and personalized Open passwords. If the master PDF was secured previously, its permissions are transferred automatically to the split PDFs.

**Example 2**

The following is another example of how you can set up a job to produce split PDFs with personalized Open passwords in four steps:

1.  Add a password field and values to a database file (*.dbf). Note the sample password field name and values in bold text.

```
%!
%%<VIDE: encoding='UTF-8'>
XGF
[  (VIPP_Pro_Pub) (Untitled-2)  ] SETPROJECT
(Password4_DM.jdt) SETJDT
(Password4_DM.dbm) STARTDBM
FirstName,LastName,Gender,Car,Addr1,City,State,Zip,Amount,Image1,Image2,Prod,Password
BOB,Kaplan,male,Camry,123 Water RD,Los Angeles,CA,90403,10,Man_1.jpg,Man_1,Vipp Pro
Publisher,Digipath1
Jane,Lillejord,female,Cadillac,33 Main St,Big City,IO,93822,20,Man_2.jpg,Man_1,Vipp Pro
Publisher,Digi
Dave,Kirk,male,Chrysler,144 Mill Lane,Little City,NM,37645,-20,Man_1.jpg,Man_1,Vipp Pro
Publisher,Digipath1
Karen,LaPointe,female,Minivan,2 High Street,New York,NY,22986,45,Man_1.jpg,Man_1,Vipp
Pro Publisher,Digi
Carlo,Sans,male,Pontiac,View Lane,Santa Barbra,CA,33234,-90,Man_2.jpg,Man_1,Vipp Pro
Publisher,Digipath1
Wang,Gary,male,Scion,1 High Street,Big City,OR,45678,0,Man_1.jpg,Man_1,Vipp Pro
Publisher,Digi
Hue,Truong,female,Solara,West View RD,Long Beach,CA,12345,10,Man_2.jpg,Man_1,Vipp Pro
Publisher,Digipath1
%%EOF%%,...........
%%EOF%%
```

2.  Create a bookmark in the database master (*.dbm) that includes the password field name. Note the sample password field name in bold text.

    `([=FirstName=] $$Password.) vsub bookmark`

    As a result of this example BOOKMARK command, a master PDF created from distillation contains bookmarks that include two fields. Note that the sample password values from step 1 are included in the second field of the bookmarks.

3.  Before you split the master PDF, go to the Job Submission Client. Next to Index File Template, click **Create Template**. As we saw from the master PDF bookmarks in step 2, there are two bookmark fields, first name and password. Set up the syntax and use your preferred bookmark field names after the first standard seven fields, FileName through FileSequence. In this example, the bookmark field name for password values is Password.

4.  Open the Security Options GUI. In the Split PDFs section, select **Enable Open Passwords**. This option enables the Job Index Field Name and its menu, which includes the bookmark field names that you specified in the Index File Template, after the FileSequence field, in step 3. Select the bookmark field name that contains the password values, which is Password in this example.

Now future split PDFs are secured with individual and personalized Open passwords. If the master PDF previously was secured, its permissions are transferred automatically to the split PDFs.

## Font Locations

When you install standard PostScript fonts, ensure that you store the fonts in the following location:

`X:\Program Files(x86)\Xerox\VIPP\xvtp\bin\normalizer\fonts`

When you install CID fonts, ensure that you store the fonts in the following location:

`X:\Program Files(x86)\Xerox\VIPP\xvtp\bin\normalizer\Resource\CIDFont`

Ensure that the CMaps are installed in the following location:

`X:\Program Files(x86)\Xerox\VIPP\xvtp\bin\normalizer\Resource\CMap`

🖉 Note:

- After you install new PostScript fonts, restart the VI eCompose server.
- Font substitution: Even if the VIPP® job has processed successfully, any font substitution can affect the appearance of the finished job.

## Assigning a Job Options File

You can assign a specific job options file to be used for generating a PDF document when you submit a VIPP® job to VIeC. The *DistillJobOptions* entry found in a Profile file is set by default to use the job options file that has been set as the default in the installed PDF generator.

When you use the built-in PDF authoring engine, the order of precedence for job options settings is as follows:

1. The highest precedence is assigned to a specific Job Options file set in a VIeC job profile, *DistillJobOptions=…* .

2. If no job profile is associated with a VIPP® job, or the profile indicates the default Job Options, *DistillJobOptions=<default>*, the default set in the `vipo.ini` file, *normalizer_options_file=…* takes effect.

3. If the `.ini` entry is set to default, *normalizer_options_file=<default>*, the Job Options file use is based on the default media selection that is set at installation:

   - If A4 media was selected, the default Job Options file is `Standard_A4.joboptions`.

   - If USLetter was selected, the default Job Options file is `Standard_USLetter.joboptions`.

To change the Adobe Job Options file used in the VI eCompose Profile, from the VI eCompose Job Submission Client menu bar, select **File > Additional Profile Options**.

Select the **Select Job Options file** option, then select **Browse**. A list of Adobe Job Options files appears. Select from the list, or create your own job options file using Adobe Acrobat.

Note: Job Option files must be located in a directory accessible to the built-in PDF authoring engine.

The file name can be entered with or without the angled brackets. For example, the Adobe Job Options file, `High_Quality_USletter.joboptions`, can be entered as `High_Quality_USletter.joboptions` or as `<High_Quality_USletter.joboptions>`.

## PDF Resources and Compatibility with APPE

The Adobe PDF Print Engine (APPE) implementations on some non-Xerox Digital Front Ends (DFEs) have difficulty printing PDF files generated by eCompose from VIPP® jobs that call out PDF files as resources. This issue is because of the way the Adobe pdfmark operator is implemented within the Adobe PDF Converter SDK (APCSDK), also known as Normalizer or Distiller.

Implementation of the Adobe pdfmark operator inserts the resource PDFs into the generated PDF as embedded referenced XObjects. The implementations of APPE on some non-Xerox DFEs do not support embedded referenced XObjects, and the PDF resources do not print correctly. eCompose can perform an optional post-processing flattening step, which extracts the embedded referenced XObjects and replaces them with equivalent form XObjects. Because not all workflows include PDF resources, not all DFEs have trouble processing embedded referenced XObjects. Additionally, this extra step incurs additional performance overhead. For these reasons, the step is not performed by default. To help with performance tuning, an optional XObject pre-check can be configured. This pre-check can detect the presence of XObjects in the job. If no XObjects are found, or none are found within a certain number of pages, eCompose does not perform the flattening step.

To enable and configure flattening of embedded referenced XObjects into form XObjects in your PDF output, manually change vipo.ini, then restart eCompose. Configure the new settings as follows:

- `perform_pdf_flatten=0`

  `pdf_flatten_requires_precheck=0`

  `flatten_precheck_pages=0`

  For these settings, do not perform the flattening, but retain the pre-flattening behavior. If the .INI settings are not present in the vipo.ini file, the pre-flattening behavior is retained.

- `perform_pdf_flatten=1`

  `pdf_flatten_requires_precheck=0`

  `flatten_precheck_pages=0`

  For these settings, perform the flattening, and do not perform the XObject detecting pre-check. Use these settings if most or all jobs used PDF resources, and the overhead of the pre-check is unwanted.

- `perform_pdf_flatten=1`

  `pdf_flatten_requires_precheck=1`

  `flatten_precheck_pages=0`

  For these settings, perform an XObject detecting pre-check for PDF XObjects. All of the pages of each job are checked. The 0 in this case is interpreted as pre-check all pages. Use these settings if the workflow was mostly a mix of smaller jobs, some using PDF resources and some not, and if the PDF resources can be located unpredictably within those jobs.

- `perform_pdf_flatten=1`

  `pdf_flatten_requires_precheck=1`

  `flatten_precheck_pages=20`

  For these settings, perform an XObject detecting pre-check for PDF XObjects, but only the first 20 pages of each job have to be checked. Use these settings for a mix of large jobs, some using PDF resources and some not, and if you know that the jobs use PDF resources in the first 20 pages.

# Using VIeC Services

This chapter contains:

VIeC and VIeC Servers can be initiated from a Windows Service, which allows the various servers to be initiated and to perform operations without requiring a manual logon. Launching and activating VIeC and VIeC servers when the system is booted up allows a hands-off VIeC system.

Use of VIeC Services depends upon the proper installation of the functionality. For a complete description of the installation process, and related background information, refer to VIeC Services Installation.

## Definitions

In the context of VI eCompose, a server is the component that performs some sort of VI eCompose work and/or function, and its associated service performs the startup and shutdown operations appropriate for a Windows Service.

### VI ECOMPOSE SERVER

A VI eCompose Server is one of the following VI eCompose components:

- FreeFlow VI eCompose
- FreeFlow VI eCompose Watched Folders Client
- FreeFlow VI eCompose Dispatch
- FreeFlow VI eCompose Secure Web Application
- FreeFlow VI eCompose Web Application

These components, once running, perform various operations related to VI eCompose.

### VI ECOMPOSE SERVICE

A VI eCompose Service is one of the following service components:

- FreeFlow VI eCompose Service
- FreeFlow VI eCompose Watched Folders Service
- FreeFlow VI eCompose Dispatch Service
- FreeFlow VI eCompose Web Job Submission Service

These components are Windows Services, which in turn are responsible for starting and stopping their associated VI eCompose Servers.

## Constraints

Windows Services must not rely on a user interface for their normal operation because for a hands-off configuration there will not be anyone logged into the machine, thus no desktop is available for displaying a user interface. Therefore, when initiated by their associated VI eCompose services, the VI eCompose servers will suppress their user interfaces entirely.

As a consequence of running in this manner, access for control or configuration of the servers is reduced. For example, the user interface for job selection and activation or rule editing and configuration for VI eCompose Dispatch is not available. Therefore, only jobs that successfully meet the qualifications of the AutoRun specifications in effect at startup will be processed.

Some remote configuration and control functionality for the VI eCompose server is available if one of the VI eCompose Web Application is running. In general, if direct interaction with a server user interface is required, it will be necessary to first shut down the associated Windows Service and then start the server application manually from the desktop. After the required interactions are performed the server application should be shut down (exited), and then the associated Window Service can be restarted.

# Using VIeC Dispatch

This chapter contains:

For information and a step-by-step guide to using VIeC Dispatch, refer to the *FreeFlow VI eCompose Workshop* VIeCWorkshop.pdf.

VIeC Dispatch provides a generic dispatch mechanism, which initiates, then monitors VIeC job post-processing by a customer-specified backend process such as email, fax, or a document repository. In this role, VIeC Dispatch can be considered middleware, because it mediates between completed VIeC jobs and the specified backend post-processing software.

Parameters and other data specific to post-processing can be embedded in the VIeC job through use of the VIPP® BOOKMARK command. The parameters and other data are extracted from the field names and values of the index file generated by VIeC and transferred to the specified backend by VIeC Dispatch. Index files for each job can be identified by the extension .csv.

VIeCD supports workflows requiring human intervention, or a hands-off workflow through the use of AutoRun and user filters. For example:

- Backends interfacing with email disbursement systems require the following:
  - Human verification or signoff of the VIeC output prior to the actual dispatch
  - Limitations on the users allowed to initiate the dispatch of such jobs

- Backends interfacing with a document repository that require a hands-off workflow in which the entire VIeC to VIeCD process runs without human intervention.

When VIeCD invokes a backend program, the parameters are extracted from a VIeC job index file on a line-by-line basis, which spawns a new instance of the backend program as a new subprocess for each invocation. VIeCD does not allow any direct interaction with the backend program over stdin/stdout during that invocation, which is a limitation that may not be suitable for interfacing with all types of backend programs. Potential incompatibilities between the VIeC post-processing to be performed and VIeCD include:

- Programs that require some form of user or programmatic interaction in their normal mode of operation, such as a Yes or No response to a file being overwritten.

- Programs that require some form of session state over a set of transactions, such as logging into a Microsoft Exchange server to perform a number of email transmissions.

- Backend solutions that involve more than one discrete operation, requiring the invocation of multiple, discrete post-processing operations; such as concatenation or other combination of the files specified in the DataFile Template prior to submission to one or more backend programs.

The VIeCD SDK, available from the Xerox SDK program, contains example code and libraries that can provide the basis for building a shim, wrapper, proxy, or facade to resolve the integration issues listed. The primary focus of the VIeCD SDK examples is centered on a separate server application acting as a session bridge between a simple client shim that is invoked by VIeCD on an index file line-by-line basis, and the backend program that has the requirement for human interaction or intervention, session state, or multiple post-processing operations.

An alternative to the client-server approach, for backend programs that do not require session state over a set of transactions, is to interpose a proxy program between VIeCD and the backend programs that are invoked once for each line in the index file. This proxy assumes the responsibility for interacting with the user if necessary, and acts in the role of a facade for a collective of discrete backend programs if multiple post-processing steps are required. This proxy communicates with VIeCD through the documented interfaces retval and stdin/stdout, as well as with the backend programs performing the actual post-processing. There are no explicit examples of this provided with the VIeCD SDK, however, the VTPDICE utility can be configured to act as a proxy between VIeCD and an otherwise VIeCD-compatible backend program. Since the VTPDICE source is provided with the VIeCD SDK, reviewing it can provide insights as to how such a proxy program can be implemented.

For more information, refer to these sections of the *FreeFlow VI eCompose Dispatch SDK User Guide*:

- Examples
- Libraries
- Utilities
- Using VIeCDICE
- Using VIeCDICE in batch mode

## VIeCD Data Flow
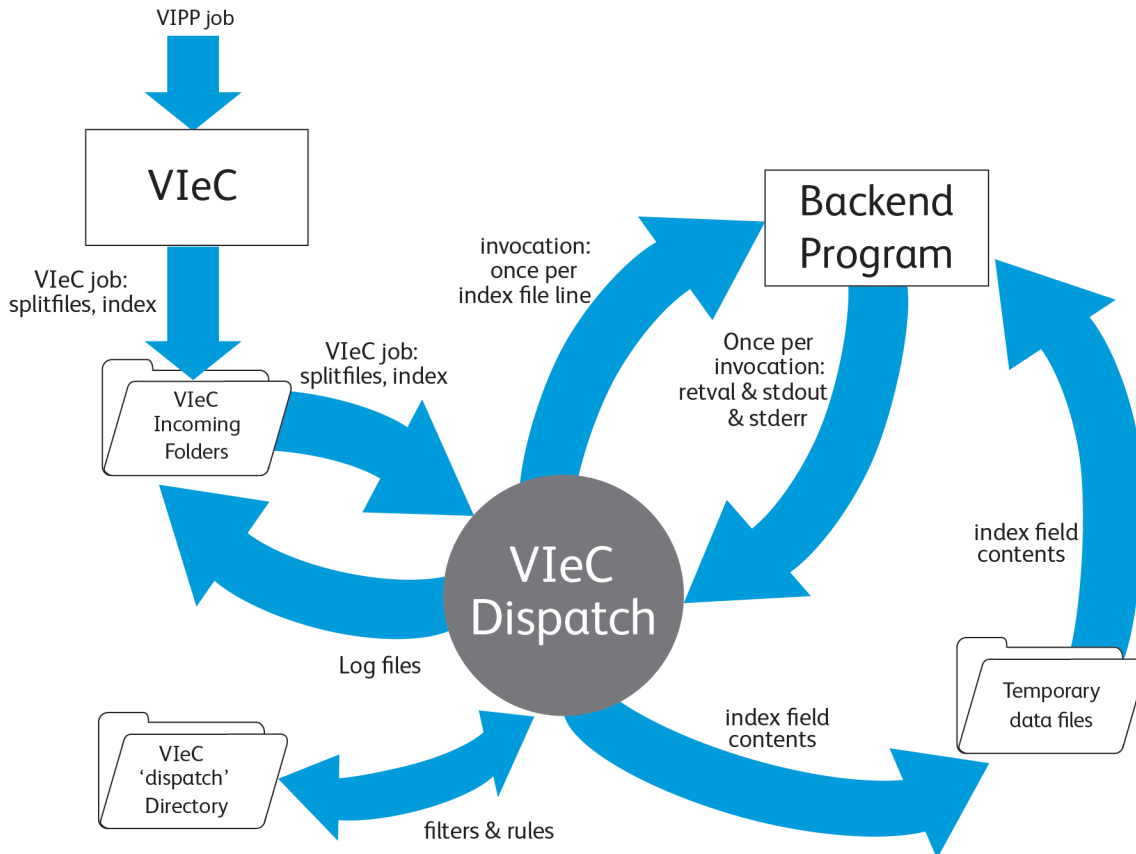


As VIeCD interfaces with backend programs it periodically inspects the VIeC IncomingFolders directories that match a specified user filter and compares the field names and contents of the first line of each VIeC job's index file against a repository of dispatch rules.

- If a completed VIeC job can be associated with a single dispatch rule, it is considered eligible for dispatch and becomes a VIeCD job.

- If the VIeCD job is either manually approved for processing or meets the user-specified criteria for a specified AutoRun filter, VIeCD then sequentially resolves and applies the appropriate dispatch rule against each line of the VIeC job index file.

In the context of a dispatch rule, each of the lines of the VIeC job index file results in a separate invocation of the customer-specified backend program, effectively becoming a subtask of the VIeCD job. Sequential execution of each subtask first (optionally) writes the contents of one or more of the index file line fields to disk and then performs an invocation of the backend program. As each subtask completes, its **retval**, **stdout** and **stderr** streams are inspected in the context of its dispatch rule to determine if a warning or error has occurred, and if so whether subtask processing should continue or be halted. The results of the subtask processing are accumulated in a log file in the same directory in which the results of VIeC job processing were stored, typically a subdirectory of the user's Incoming directory.

## VIeCD IncomingFolders Filters

VIeCD monitors the VIeC IncomingFolders directory in the context of a specified IncomingFolders filter. The default setting allows VIeCD to monitor all IncomingFolders for all VIeC users. However, VIeCD can be set to process jobs for a particular set of users, and/or for a particular set of IncomingFolders, as only the specified IncomingFolders of the VIeC users that match the specified user filter are considered for VIeCD processing.

## Eligibility: CommandTemplates, RuleVars, and the DispatchRule Fieldname

Completed VIeC jobs that match the IncomingFolders filter are evaluated to determine the file eligibility for processing by VIeC Dispatch. A VIeC job is deemed eligible for processing by VIeC Dispatch if the file can be matched to a single dispatch rule. The CommandTemplate section of a dispatch rule determines which backend program is invoked, and with which parameters.

### INDEX FILE

To determine eligibility, VIeC Dispatch inspects the index file field names of each VIeC job, then compares them against each CommandTemplate section of the available dispatch rules. In Example 1, if none of the CommandTemplates for any dispatch rule contains the mailto field, the VIeC job is ineligible because no dispatch rule can be applied. If the index file field names of a VIeC job can be matched with the CommandTemplate of exactly one dispatch rule, it is considered eligible.

**Example 1**

In this example the index file field names and single dispatch rule CommandTemplate of the following VIeC jobs are:

```
...,"Pages","FileSequence","mailto"
```

```
blat c:\bodytemp.txt -t $mailto
```

The VIeC job is eligible for VIeCD processing if the dispatch rule above is the only one with a CommandTemplate containing the single mailto field.

Alternately, the VIeC job becomes ineligible for processing if there was another rule that contains the following CommandTemplate variable information, because more than one dispatch rule can then be applied:

```
splat -x $mailto
```

**Example 2**

This example assumes:

- Two VIeC jobs, each containing one of these index file field names:

  ```
  ...,"Pages","FileSequence","mailto"
  ```

  ```
  ...,"Pages","FileSequence","mailto","cc"
  ```

- Exactly two dispatch rules, each containing one of the following as their CommandTemplate:

  ```
  blat c:\bodytemp.txt -t $mailto
  ```

  ```
  blat c:\bodytemp.txt -t $mailto -c $cc
  ```

In this example, both VIeC jobs are eligible for processing because VIeCD recognizes that the first VIeC job cannot be used with the second dispatch rule, as there is no field named cc in the first job's index file, so the first dispatch rule is applied for the first VIeC job. VIeCD also recognizes that the second VIeC job cannot be used with the first dispatch rule, as there is no field named cc in the CommandTemplate of the first dispatch rule, so the second dispatch rule is applied to the second VIeC job.

## RULEVARS

In addition to the index file field names defined by a VIeC job, a Dispatch rule can define additional fields and values to be applied to the CommandTemplate. A typical requirement is a Dispatch rule whose CommandTemplate requires a password to execute, because it is undesirable for the password to be included in the VIeC job. Such additional fields and their values can be specified in the RuleVar section of the Dispatch rule.

Example 3

In this example, the Dispatch rule CommandTemplate and the single Dispatch rule with the user field present are as follows:

```
foo -user $user -password $password
```

```
...,"Pages","FileSequence","user"
```

In this event, the VIeC job is ineligible for processing because there are no password fields among the index file field names.

Example 4

The VIeC job in Example 3 becomes eligible for processing when the Dispatch rule has a **RuleVar** entry for the password field, as follows:

```
password=mypassword
```

In this event, the password field of the CommandTemplate is supplied by the Dispatch rule **RuleVar** definition.

## RESERVED INDEX FILE FIELD NAME

The final mechanism used within VIeC Dispatch for choosing among otherwise ambiguous dispatch rules for a particular VIeC job, involves the use of a reserved index file field name. If the VIeC job has among the index file field names the reserved field name `DispatchRule`, the content of that field for the first line of the index file is matched against the rule names of the set of Dispatch rules found to be ambiguous.

Example 5

Ambiguity is resolved and the Dispatch rule is applied to the VIeC job when the following conditions are met in a job with index file field names of `mailto` and `DispatchRule`:

- The job is compared to the two rules containing blat and splat described in Example 1, because of the mailto fieldname.

- The contents of the reserved field name, `DispatchRule`, for the first line of the index record is `SMTP email`

- One of the dispatch rules has `SMTP email` as the Dispatch rule name

## AutoRun Filters

After a VIeC job is deemed eligible for processing by VIeCD, the file is compared to the currently specified AutoRun filter. AutoRun filters allow the specification of a particular set of VIeC users and a particular set of IncomingFolders for those users, to determine which eligible VIeC jobs are processed automatically by VIeCD. The default setting is to disallow AutoRun for all users. To process files, select each job manually.

## Processing

When VIeC Dispatch processes a VIeC job, it reads the index file for the VIeC job one line at a time, and applies field values in the context of the applicable dispatch rule. Each application results in a separate invocation of the backend program specified in the dispatch rule CommandTemplate. Each of the invocations, one for each line in the VIeC job index file, is considered a subtask of the job.

The following occurs for each line of the index file, until a halt-on-warning or halt-on-error condition is detected, or there are no lines left in the index file to process:

- If any of the field names are present in the DataFileTemplate section of the dispatch rule, the field name contents are written to disk as specified. The files written are considered temporary, and the contents are considered valid only for the duration of the particular invocation of the backend program specified in the CommandTemplate.

- The field names in the CommandTemplate are resolved by substituting the values from the *RuleVars* and from the values extracted from the pertinent line of the index file. The result is the command and parameters intended for the invocation of the backend program. The command and parameters are submitted to the operating system for execution as a subprocess of VIeC Dispatch.

- When the subprocess for the backend program completes, VIeCD interprets the **retval**, **stdout** and **stderr** streams in the pertinent sections of the dispatch rule to determine if a warning or error situation has been triggered, and if processing of the remaining lines of the index file continues.

## VIeCD Job Life Cycle

During its life cycle a VIeCD job can transition between any of the following states:

- ineligible

- rule conflict

- eligible

- pending

- current

- held

- complete

These states are reflected in the Current Job and tabbed sections of the main VIeC Dispatch GUI and they are described in more detail following.
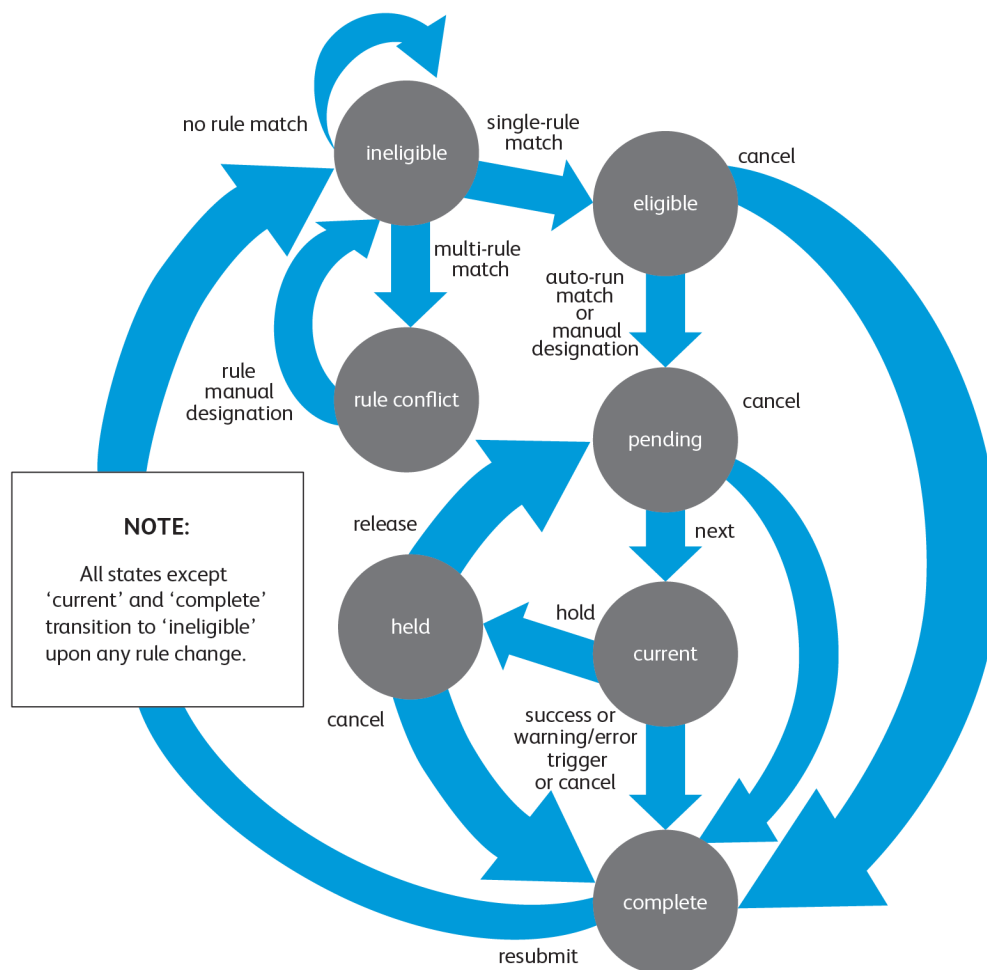
### INELIGIBLE

All VIeC jobs that make it past the IncomingFolders filter result in the instantiation of a VIeCD job with an initial state of ineligible, making it the start state of the VIeCD job lifecycle. VIeCD jobs entering the ineligible state are tested immediately for eligibility. The first index file record field names are evaluated against the CommandTemplate and RuleVar of each available dispatch rule. If the field names can be associated with a dispatch rule, the following actions occur:

| | |
|---|---|
| Only one dispatch rule | The VIeCD job transitions to the eligible state. |
| More than one dispatch rule | The job transitions to the rule conflict state. |
| No dispatch rules | The job remains in the ineligible state. |

### RULE CONFLICT

Jobs in the rule conflict state have more than one applicable dispatch rule. Transition out of the rule conflict state occurs in either of the following conditions:

- The rule to be applied is selected manually.

- Any dispatch rule is changed or added.

In either case, the job transitions back to the ineligible state where the file is re-evaluated. After the file is returned to the ineligible state:

- If a dispatch rule was manually selected, the job transitions to the eligible state.

- If no dispatch rules can be applied, the job can remain ineligible.

- If more than one applicable dispatch rule remains, the job can transition back to the rule conflict state.

### ELIGIBLE

Jobs in this state are eligible for processing but are not yet pending. A job remains in an eligible state until it is manually selected by the user for processing or meets the criteria of the current AutoRun filter. When the job meets the AutoRun filter criteria it then automatically transitions to the pending state.

Canceling a job in an eligible state causes it to transition to a complete state with a status of cancel.

### PENDING

Jobs in this state will be processed in the order shown in the VIeC Dispatch Pending display, topmost first. The order of the pending jobs can be changed using the GUI.

Canceling a job in a pending state causes it to transition to a complete state with a status of cancel.

### CURRENT

A job in this state is being processed. There can be only one job in this state.

Holding a current job causes it to temporarily suspend processing (in a graceful manner) and move into the held state.

Canceling a job in a current state causes it to transition to a complete state with a status of cancel.

### HELD

A held job has suspended processing. Releasing the job causes it to transition to a pending state to await further processing.

Canceling a job in a held state causes it to transition to a complete state with a status of cancel.

### COMPLETE

This is the end state of the VIeCD job lifecycle. Completed jobs have one of these statuses:

- success
- warning
- failure
- cancel

The resubmit transition is supplied for convenience. Resubmitting a job clears all accumulated state, tracking, etc., for that job and returns it to the ineligible state, allowing it to be reprocessed as if it had just come into the system for the first time.

## Dispatch Filters

A filter file can have up to 100 user-folder pair entries, with a maximum of 2048 characters for each entry. Only one AutoRun filter and one IncomingFolders filter can be active during a current invocation of the application. The active filters can be set with the Set Active Filter button found in the Editing dialog panel. The button is grayed-out when the active filter is displayed.

All filter files require a file extension of `.filter` and require placement in the corresponding directory shown so that the application finds and makes the files available for editing:

- The AutoRun filter files are required in this directory: `X:\ProgramFiles(x86)\Xerox\VIPP\xvtp\dispatch\filters\AutoRun\`

- The IncomingFolders filter files are required in this directory: `X:\ProgramFiles(x86)\Xerox\VIPP\xvtp\dispatch\filters\IncomingFolders\`

  Where `X` is the disk drive where VIeC was installed initially.

To edit Dispatch Filter files, click the **Edit** menu, then select **AutoRun** or **IncomingFolders** filters.

One default AutoRun filter and one default IncomingFolders filter are provided with this installation.

- Always use the GUI editing facilities to modify filter files. Manual edits can introduce errors in the file that can result in unpredictable behavior.

- All filter, file, and rule name fields are case-sensitive. A filter entry of `all` or `none` for the user or folder key disables all other options from being selected. To change an `all` or `none` setting, before selecting a new option, delete or clear the entry.

The following are special filter entries that are allowed as the only single entry in a filter. No other entries are allowed with a special filter entry in the filter file.

```
user=all
folder=all
or
user=all
folder=none
or
user=all
folder=rule-based
```

✎ Note: `All`, `none`, and `rule-based` are reserved words used for special filter entries. Do not use these words to define a user name or an incoming folder name. If you use the reserved words, VIeC Dispatch interprets the reserved words as special filter entries rather than a user name or an incoming folder name, and unexpected results can occur.

## Dispatch Rule Files

All rule files require a file extension of .rule and require placement in the `X:\ProgramFiles(x86)\Xerox` `\VIPP\xvtp\dispatch\rules\` directory so that the application can find the files for editing. In the directory path, `X` is the disk drive where VIeC was installed initially.

To edit Dispatch rule files, click the **Edit** menu, then select **Dispatch Rules**. Some sample rule files are provided with this installation. You can use the sample rule files as a template to create customized rule files.

🖉 Note:

- Manual edits can introduce errors.

- To modify rule files, always use the GUI editing facilities. Manual edits can introduce errors in the file that can result in unpredictable behavior.

- All filter, file, and rule name fields are case-sensitive.

The following are defaults and limits imposed on Dispatch Rule settings.

| KEY NAME | DEFAULTS | MAXIMUM LENGTH OR VALUE |
|---|---|---|
| RuleName | Rule Filename | 75 characters<br><br>Invalid characters:<br><br>\ / : * ? < > \| ' ' ~ ! $ ^ & , ; % # = + @ " |
| RuleVars | <empty> (optional) | Variable Name: 25 characters<br><br>Invalid characters: \ / : * ? < > \| ' ' ~ ! $ a & , ; % # = + @ " and blank spaces<br><br>Variable Value: 70 characters<br><br>Invalid characters: * ? < > \| ' ' ~ ! a & , ; = +<br><br>There is a maximum of 25 Name-Value sets |
| DataFileTemplate | <empty> (optional) | Field Name: 25 characters<br><br>Invalid characters: \ : * ? < > \| ' ' ~ ! $ a & , ; % # = + @ " and blank spaces<br><br>Destination File Path: 90 characters<br><br>Invalid characters: * ? < > \| ' ' ~ ! a & , ; = + @ " and blank spaces<br><br>There is a maximum of 20 FieldName and FilePath sets |
| CommandTemplate | NO default and NOT optional must | 1024 characters |

| KEY NAME | DEFAULTS | MAXIMUM LENGTH OR VALUE |
|---|---|---|
| | be defined by user | The CommandTemplate definition can include the optional NOOP_ CMD_PROC flag to indicate that Dispatch should skip the command invocation for each index file record, for example, a no-op command. |
| | | The NOOP_CMD_PROC flag can be used instead of a custom no-op process or executable when the actual work is postponed until the execution of a custom process defined in the PostProcessCommandTemplate entry. |
| | | Example: |
| | | CommandTemplate=NOOP_CMD_ PROC $(InputFile) $(Bookmark)... |
| TimeOut | | 32,400 seconds or about 9.0 hours |
| LogStdOut | true (boolean) | |
| LogStdErr | true (boolean) | |
| FailOnWarning | false (boolean) | |
| FailOnError | true (boolean) | |
| RetValTriggerTrumpsStdOutStd Err | true (boolean) | |
| RetValTrigger | <empty> (optional) | 100 characters for each of the three integer triggers |
| | | Valid characters: 0 1 2 3 4 5 6 7 8 9 < = > - | * and blank spaces |
| StdOutTrigger | <empty> (optional) | 256 characters for each of the three string triggers |
| | | Invalid characters: \ / ; |
| StdErrTrigger | <empty> (optional) | 256 characters for each of the three string triggers |
| | | Invalid characters: \ / ; |
| AutoRun | false (boolean) | |

| KEY NAME | DEFAULTS | MAXIMUM LENGTH OR VALUE |
|---|---|---|
| RunPostProcessCmd | false (boolean) | |
| PostProcessCommandTemplate | <empty> (optional) | 1024 characters |

The Dispatch Rule entries are defined as follows:

### RuleName

The user-specified name by which the rule is identified to the user. For example, `Email via Exchange` or `Submit to DocuShare`.

### RuleVars

A set of name-value pair definitions that define the specified field names for use in the CommandTemplate. When a RuleVars name is the same as a field name encountered in a VIeCD job splitfile index record, the field name contents of the splitfile index record takes precedence over the RuleVars value. For example:

`RuleVars=(profileName="Bill Walker-Test2" | password=wazoo)`

The RuleVars entry can be empty, indicating that no RuleVars name or value pairs are defined for the given Dispatch rules.

### DataFileTemplate

The template from which the data file preprocessing steps are defined.

The preprocessing steps specify the writing to disk of zero or more files containing the contents of one or more splitfile index fields. This action is required to support external programs like some batch email clients, where the body of an email message is specified as a file name on the client command line.

The template consists of a set of rules internally delimited by the `|` character and delimited interstitially by the `;` character of the form:

`FieldName|DestinationFilePath|deleteOnExit;...`

The `FieldName` entry is the name of a field in the splitfile index. For example, the field names `body` or `mailtext`. The path resolves to the full path of a destination file. The boolean `deleteOnExit` indicates if the destination file is deleted by VIeCD upon successful processing of the subtask. For example:

`DataFileTemplate=(body | C:\Program Files (x86)\Xerox\VIPP\xvtp\bin\bodytemp.txt | 1)`

The contents of the `body` field of the splitfile index record is written to the file `c:\Program Files (x86)\Xerox\VIPP\xvtp\bin\bodytemp.txt`, which is then deleted after the execution of the rule command. For example:

`DataFileTemplate=(body | C:\Program Files (x86)\Xerox\VIPP\xvtp\bin\ $(destFilename) | 0)`

The `destFilename` entry is a field of the splitfile index record having the value `jones.txt`, results in the contents of the body field of the splitfile index record being written to the file `c:\Program Files (x86)\Xerox\VIPP\xvtp\bin\jones.txt`. The file is not deleted after the execution of the rule command.

A DataFileTemplate can be empty, indicating that no file creation is to take place.

### CommandTemplate

The template from which the command that is passed to the operating system that represents the execution of the subtask is derived. For example:

```
blat C:\Program Files (x86)\Xerox\VIPP\xvtp\bin\bodytemp.txt -t $mailto
```

The `mailto` entry is a field of the splitfile index record or RuleVars name having the value jones@ip.com results in the file `C:\Program Files (x86)\Xerox\VIPP\xvtp\bin\bodytemp.txt` being mailed to jones@ip.com using the SMTP batch tool, blat, if blat.exe was somewhere on the execution path of the VIeCD process.

The command template is preprocessed to produce the actual command string that is passed to the OS. Command template fields that begin with the reserved character `$` are substituted by the respective splitfile index record field or RuleVars value. VIeCD assumes that a single target executable is called when the command is executed. Multiple-step executions are not supported by VIeCD. If a multiple-step subtask is needed for a given VIeCD job, the subtask is required to call a single batch process, which is responsible for executing the multiple steps and reporting status back to VIeCD.

A command template cannot be empty and to be valid, requires at least one index file record fieldname reference, beginning with the reserved character `$`. Otherwise, the Dispatch rules do not associate with any job when evaluating the job eligibility against the Dispatch rules CommandTemplate definition.

### TimeOut

Specifies the length of time in seconds between when the OS command is executed and a TimeOut error is declared. A TimeOut value of 0 indicates an infinite waiting period. An OS command that triggers a TimeOut should be terminated by VIeCD and treated as having returned with an error return value.

### LogStdOut

A boolean `.INI false: 0,` or `true: 1` indicates if the stdout of the executing OS command is appended to the log file for the VIeCD job associated with this Dispatch rule.

### LogStdErr

A boolean `.INI false: 0,` or `true: 1` indicates if the stderr of the executing OS command is appended to the log file for the VIeCD job associated with this Dispatch rule.

### FailOnWarning

A boolean `.INI false: 0,` or `true: 1` indicates if detection of a warning during the execution of the OS command causes the VIeCD job to halt processing and transition from `current` to `complete`, with a status of `failure` at the record that generated the error.

### FailOnError

A boolean `.INI false: 0,` or `true: 1` indicates if detection of an error during execution of the OS command causes the VIeCD job to halt processing and transition from `current` to `complete`, with a status of `failure` at the record that generated the error.

### RetValTriggerTrumpsStdoutStderr

If true, `.INI true: 1` indicates that the RetValTrigger value takes precedence over any return value indication returned by StdOutTrigger or StdErrTrigger. If false, 0, the return value indication from StdOutTrigger or StdErrTrigger takes precedence over the value returned by RetValTrigger.

### RetValTrigger

The definition of the integer values returned by the executed OS command are interpreted in the context of success, warning, or failure. These values are presented as a set of rules delimited internally by the `|` character and

delimited interstitially by the ; character of the form:

```
trigger(s);trigger(s);trigger(s)
```

The first set, the highest precedence, defines the return values that indicate success, the second set defines the return values that indicate warning, and the third set and lowest precedence defines the return values indicating failure. In a trigger definition, the asterisk character * indicates any value that causes a trigger, except those values defined in a trigger with higher precedence. An empty value indicates that no triggers exist at that precedence. For example:

```
RetValTrigger=(0 ; ; *)
```

A 0 return value indicates success, no values indicate warning, and any value other than those values defined for success or warning indicates failure.

Within a trigger definition, multiple sets of conditions can be separated by the | character, for example:

```
RetValTrigger=(0 ; 1 | 2 ; <0)
```

A 0 return value indicates success, a 1 or 2 value indicates a warning, and any negative value indicates failure.

Within a trigger definition, the following operators are recognized: <, <=, >, > =.

For example:

```
RetValTrigger=(0 ; >0 ; <0)
```

A 0 return value indicates success, values greater than 0 indicate a warning, and values less than 0 indicate failure.

A RetValTrigger definition can be empty, indicating that return values are ignored. The OS command is assumed to have completed with success, regardless of the return value, unless surpassed by the stdout or stderr triggers.

Any value returned by the OS command not covered by the RetValTrigger definitions is treated as a warning return value.

If RetValTrumpsStdOutStdErr is false, the RetValTrigger return value indication is surpassed by the values detected using a StdOutTrigger or a StdErrTrigger. A surpassed RetValTrigger return value indication does not preclude the RetValTrigger value from being considered for inclusion in the log file.

### StdOutTrigger

The definition of substring values whose presence in the stdout stream emitted by the executed OS command, are interpreted in the context of success, warning, or failure.

These values are presented as a set of rules delimited internally by the | character and delimited interstitially by the ; character of the form:

```
trigger(s);trigger(s);trigger(s)
```

The first set and highest precedence defines the substrings whose presence indicates success, the second set defines the values that indicate warning, and the third set and lowest precedence defines the values indicating failure. In a trigger definition, an empty value indicates that no triggers exist at that precedence. For example:

```
StdOutTrigger=(; ; "Error:")
```

The presence of the substring `Error:` in the OS command stdout indicates an error.

For example:

```
StdOutTrigger=(; "Warning:" ; "Error:" | "Fatal:")
```

The presence of the substring `Warning:` in stdout indicates a warning, while the presence of `Error:` or `Fatal:` indicates an error.

A StdOutTrigger definition can be empty, indicating that the contents of the stdout stream is ignored for purposes of determining the return status of the OS command.

A StdOutTrigger return value indication is surpassed by those detected using a StdErrTrigger, or by a RetValTrigger return value indication if RetValTrumpsStdOutStdErr is true. A surpassed StdOutTrigger return value indication does not preclude the StdOutTrigger value from being considered for inclusion in the log file.

### StdErrTrigger

The definition of substring values, whose presence in the stderr stream emitted by the executed OS command, are interpreted in the context of success, warning, or failure.

The values are presented as a set of rules delimited internally by the `|` character and delimited interstitially by the `;` character of the form:

```
trigger(s);trigger(s);trigger(s)
```

The first set and highest precedence, defines the substrings whose presence indicates success, the second set defines the values that indicate warning, and the third and lowest precedence defines the values indicating failure. In a trigger definition, an empty value indicates that no triggers exist at that precedence. For example:

```
StdErrTrigger=(; ; "Error:")
```

The presence of the substring `Error:` in the OS command stderr indicates an error.

For example:

```
StdErrTrigger=(; "Warning:"; "Error:"|"Fatal:")
```

The presence of the substring `Warning:` in stderr indicates a warning, and the presence of `Error:` or `Fatal:` indicates an error.

A StdErrTrigger definition can be empty, indicating that the contents of the stderr stream is ignored for purposes of determining the return status of the OS command.

A StdOutTrigger return value indication is surpassed by a RetValTrigger return value indication if RetValTriggersStdoutStdErr is true. A surpassed StdErrTrigger return value indication does not preclude the StdErrTrigger value from being considered for inclusion in the log file.

### AutoRun

If the current AutoRun filter is set to give rules control of automatic job processing, the AutoRun entry in Dispatch rules allows you to control automatic processing of eligible jobs on a rule-by-rule basis. For example, if the AutoRun filter entry for folder has been set to a rule-based value, automatic job processing is controlled by the setting of the AutoRun rule entry.

> Note: If the AutoRun filter has not been set to yield control to the Dispatch rules, the AutoRun setting in Dispatch rules is ignored. The AutoRun filter setting acts as a global on and off switch that can take or give control of the automatic processing of all jobs.

### RunPostProcessCmd=[true/false]

This boolean entry indicates whether to execute the command string found in the PostProcessCommandTemplate entry if not empty. Use this entry to turn the post-processing command on and off without having to remove or comment-out the actual command string.

**PostProcessCommandTemplate=(...command string...)**

This optional free-format rule entry can specify a command string that is executed after the job has been processed successfully by VIeC Dispatch. Use this entry to perform post-processing tasks on a per-job basis. The post-processing command string has a maximum length of 1024 bytes, and can include field name variable references and built-in macros that are resolved before the command is sent for execution.

> Note: After the command string is sent to the operating system for execution, VIeC Dispatch does not wait for success or failure. VIeC Dispatch sends the post-processing command string to execute, then moves on to process the next job.

### SYSTEM BUILT-IN MACROS

Use these built-in macros as variables in a rule file for DataFileTemplate, CommandTemplate, and PostProcessCommandTemplate definitions:

| | |
|---|---|
| PDF_DIR_PATH | Full path of the directory location where the distilled PDF file a ssociated with the job can be found. |
| PDF_FILE_PATH | Full path of the individual split PDF file associated with the index file record currently being processed. |
| PDF_FILE_NAME | File name only of the unique split PDF file associated with the index file record currently being processed. |
| USER_NAME_ID | User ID associated with the job being processed. |
| JOB_NAME_ID | Job Name ID associated with the job being processed. |
| JOB_INDEX_FILE_NAME | Name of the Job Index file being used by Dispatch when processing a job. |
| SUBMIT_JOB_FILE_PATH | The full path of the submission file associated with the job being processed. |
| SUBMIT_JOB_FILE_ NAME | Name of the submission file associated with the job being processed. |
| JOB_INDEX_FILE_PATH | The full path to the Job Index file being used by Dispatch when processing a job. |
| SPLIT_PDF_FILE_NAME | File name only of the unique split PDF file associated with the index file record currently being processed. This macro is equivalent to the existing PDF_FILE_NAME macro. |
| SPLIT_PDF_FILE_PATH | Full path of the individual split PDF file associated with the index file record currently being processed. This macro is equivalent to the existing PDF_FILE_PATH macro. |
| MASTER_PDF_FILE_ NAME | File name only of the Master PDF file that was used as input to generate all the split PDF files. |
| MASTER_PDF_FILE_PATH | Full path of the Master PDF file that was used as input to generate all the split PDF files. |
| JOB_FILE_BASE_NAME | File name only, without extension, of the Master PDF file that was used as input to generate all the split PDF files. |
| MASTER_PDF_PAGES_ | The number of pages of the Master PDF file. |

DISTILLED

| | |
|---|---|
| MASTER_PDF_PAGES_ SPLIT | The number of pages split from the Master PDF file. |
| MASTER_PDF_FILES_ CREATED | The number of split files created from the Master PDF file. |
| MASTER_PDF_STATUS_ MSG | Contains any warnings that occurred when generating the Master PDF file or the split PDF files. |

> 🖉 Note: Use of this macro requires the .ini configuration setting 'enable_ distillation_trace_log' be set to either 2 or 4.

These built-in macros can be used as variables in a job profile for only the NotificationCommandStr= definition:

| | |
|---|---|
| JOB_DIR_PATH | Full directory path of the destination job folder. |
| JOB_FOLDER_NAME | Name of the destination job folder. |
| JOB_FILE_NAME | Name of the job submission file. |
| JOB_FILE_PATH | Full path of the original submission file in the context of the submitting client. |
| JOB_USER_NAME | Name of the VIeC User associated with the job. |
| JOB_SERVER_NAME | Name of the VIeC Server, that is the Computer Name, that picks up the job for processing. |
| JOB_STATUS_MSG | Job status string that contains either the error string or warning message, depending on the trigger set up for the notification command. |

### RULE FILE VARIABLE REFERENCES

The DataFileTemplate, CommandTemplate, and PostProcessingCommandTemplate definitions allow the use of variable references. The variable names used can be field names from the header of the job index file, variable names defined within the RuleVars file, or system built-in macros. The reserved character $ is used to indicate a variable reference. Only the following two syntax forms are supported:

`$varname`

`$(varname)`

The following are examples only and are not intended to work in all conditions.

These CommandTemplate definitions have variable references:

```
body | C:\xvtp\bin\$(TempFolder)\bodytemp.txt | 0
body | C:\xvtp\bin\temp\$(BodyFileName) | 0
```

These DataFileTemplate definitions have variable references:

```
blat C:\xvtp\bin\$(TempFolder)\bodytemp.txt -t $mailto
blat C:\xvtp\bin\$(TempFolder)\$(BodyFileName) -t $mailto
xcopy $(OutputDir)\$(FileName) C:\xvtp\dispatch\MyArchiveFolder /y
xcopy $(PDF_FILE_PATH) C:\xvtp\dispatch\MyArchiveFolder /y
```

## Special Character Sequences in CSV and XML Index Files

All new line characters (<CR> and <LF>) found in bookmark fields extracted from the distilled PDF file are first converted to a special sequence of characters before placing the bookmark string in the CSV and XML index files. The character sequence &#x0A; is used to replace all new line characters found in the bookmark fields. This allows the index files to carry line break information within the bookmark contents without breaking the format of the index file record, and to pass the information to any utility being used to parse and extract information from the index file records.

All double quotes (") found in the bookmark fields are also converted to a special sequence of characters before placing the bookmark string in the CSV file. For consistency, the standard XML character sequence &quot; is used to replace all double quotes in CSV index files.

The following INI entries can be modified to change the default character sequences used to convert new lines and double quotes in a CSV index file:

- csv_newline_char_seq=(&#x0A;)

- csv_dquotes_char_seq=(&quot;)

> Note: The settings apply only to CSV index files. The settings do not change the character sequences used in the XML index file. When you redefine the character sequences, ensure that you do not use invalid characters. Invalid characters can corrupt the format of the CSV file. Do not use the following characters in the special character sequences: < > ' " and blank spaces.

> **Caution:** Some sequences that appear valid can prevent the index file from being processed by VIPP®. For example, the following sequence appears as if it is a placeholder delimiter for the VSUB command: [=... =]. Do not modify the default character sequences unless it is absolutely necessary.

# VIeC Dispatch Window

The VI eCompose Dispatch window provides access to Dispatch functions.

🖉 Note: All filter, file, and rule name fields are case-sensitive.

The upper portion of the Dispatch window consists of these elements:

| | |
|---|---|
| Title bar | Displays the name of the program and the computer name. Contains the standard minimize, maximize, and cancel buttons. |
| Menu bar | Provides access to the functions that are described in more detail later in this topic: |
| | File menu |
| | Pause Dispatcher |
| | Save Logfile |
| | Clear Logfile |
| | Hide Dispatcher |
| | Edit menu |
| | IncomingFolders Filters |
| | AutoRun Filters |
| | Dispatch Rules |
| | Trace menu |
| | Clear Display |
| | Job Selection Loop |
| | Job Selection Trace |
| | Help menu |
| | Help |
| | About |

**Current Job**

Lists the job being processed.

**Hold**

Places the current job on hold.

**Cancel**

Cancels processing of the current job.

**Log file window**

Contains the logged information for the VIeC jobs being processed.

The lower portion of the VI eCompose Dispatch window consists of these six tabs, with corresponding windows and buttons for each function:

- Pending Jobs
- Eligible Jobs
- Conflict Jobs
- Ineligible Jobs
- Held Jobs
- Completed Jobs

### PENDING JOBS

When the Pending Jobs tab is selected, VIeC produces a window containing a list of all VIeC jobs pending processing.

When any entry in the Pending Jobs list is highlighted, these buttons can be used to alter how the selected job will be processed:

**Move Up**

Moves the selected file up one position on the Pending Jobs list. This affects the processing order of the jobs.

**Move to Top**

Moves the selected file to the top of the Pending Jobs list. The selected job will be the next job processed.

**Move Down**

Moves the selected file down one position on the Pending Jobs list. This affects the processing order of the jobs.

**Move to Bottom**

Moves the selected file to the end of the Pending Jobs list. The selected job will be the last job processed.

**Hold Job**

Holds the selected job. Once placed on hold, the job appears on the Held Jobs window until you choose to release or cancel that job.

**Cancel Job**

Cancels the selected job. Once selected for cancellation, the job is removed from the list and placed in the Completed Jobs queue with a "Cancelled" status.

### ELIGIBLE JOBS

When the Eligible Jobs tab is selected, VIeC produces a window containing a list of all VIeC jobs eligible for processing.

When any entry in the Eligible Jobs list is highlighted, these buttons can be used to alter how the selected job will be processed:

**Preview Job File**

Previews the PDF file. Adobe Acrobat or Reader is launched and the original, unsplit, distilled master PDF will be previewed. This option is used to visually inspect and verify the job before submitting it to the Pending Jobs queue.

**Submit Job**

Submits the selected job for processing. Once selected the job will appear on the Pending Jobs window.

**Hold Job**

Holds the selected job. Once placed on hold, the job appears on the Held Jobs window until you choose to release or cancel that job.

**Cancel Job**

Cancels the selected job. Once selected for cancellation, the job is removed from the list and placed in the Completed Jobs queue with a cancelled status.

### CONFLICT JOBS

When you select the Conflict Jobs tab, VIeC produces a window that contains a list of all VIeC jobs with conflicts. More than one Dispatch rule can be applied to Conflict jobs.

When any entry in the Conflict Jobs list is highlighted, to alter how the selected job is processed, select one of the options:

- **Select dispatch rules**: This option allows you to select the appropriate Dispatch rules from a list.

- **Cancel Job**: To cancel the selected job, select **Cancel Job**. The job is removed from the list and placed in the Completed Jobs queue with a `Cancelled` status.

- **Ineligible Jobs**: To view all ineligible jobs, select **Ineligible Jobs**. When you select the Ineligible Jobs tab, VIeC produces a window containing a list of all ineligible VIeC jobs. No existing Dispatch rules could be applied at the time the job was accessed by VIeC Dispatch, so the job became an ineligible job. When any entry in the Ineligible Jobs list is highlighted, to alter how the selected job is processed, select an option:

    - **View IndexRecord Field Names**: To inspect the field names in the CSV index file header to determine why no existing Dispatch rules can be applied, select **View IndexRecord Field Names**.

    - **Cancel Job**: To cancel the selected job, select **Cancel Job**. The job is removed from the list and placed in the Completed Jobs queue with a `Cancelled` status.

    - **Held Jobs**: When you select the Held Jobs tab, VIeC produces a window containing a list of all VIeC jobs that are on hold. When any entry in the Held Jobs list is highlighted, to alter how the selected job is processed, select an option:

        - **Release Job**: Releases the selected job from the Hold status. The job eligibility is reevaluated against the current set of Dispatch rules, which can result in the job moving to the Ineligible, Eligible, or Conflict state.

        - **Cancel Job**: Cancels the selected job. The job is removed from the list and placed in the Completed Jobs queue with a `Cancelled` status.

### COMPLETED JOBS

When the Completed Jobs tab is selected, VIeC produces a window that contains a list of all completed VIeC jobs.

When any of the entries in the Completed Jobs list is highlighted, to alter how the selected job is processed, select

one of the following options:

- **Review Job Status**: This option provides the status of the selected job.

- **Resubmit Job**: This option resubmits a job for processing. The job eligibility is reevaluated against the current set of dispatch rules. The job can move to the Ineligible, Eligible, or Conflict state.

- **Preempted Job**: When multiple Dispatch servers are running, it is normal for an occasional preemption to occur. Preempted means that one Dispatch server was considering a job for processing, but that job was picked up and processed up by another Dispatch server before the first server could complete the job.

## FILE MENU

The File menu provides drop-down access to these functions:

### Hide Dispatcher

Minimizes the VIeC Dispatch window. The window can be reopened by double-clicking the VIeC Dispatch icon on the lower right menu bar.

### Pause Dispatcher

Pauses the VIeC Dispatch program. To restart VIeC Dispatch, remove the check mark. The status of the program is displayed in the upper logfile window.

### Save Logfile

Saves the current logfile to the disk. You will be prompted to browse for the directory in which to save the logfile.

### Clear Logfile

Clears the entries that appear on the logfile window.

## EDIT MENU

The Edit menu provides drop-down access to the following functions.

### IncomingFolders Filters

To select and edit the IncomingFolders Filters file, select **IncomingFolders Filter File**. The filters are used to decide which VIeC jobs make it into VIeC Dispatch for evaluation against the Dispatch rules.

The following options are available in the IncomingFolders Filter File window:

- **Filter Filename**: Contains a list of all defined IncomingFolders Filter files found in the VIeC Dispatch Server IncomingFolders directory. Highlight the file name that you want to edit. The entries in the highlighted filter file appear in the lower dialog.

- **Set Active Filter**: Sets the highlighted filter to active status to get potential VIeC jobs into the dispatcher.

- **Delete File**: Deletes the selected filter file from the list and from the disk.

- **Edit**: Produces a dialog that lists user names and incoming folder names for the selected filter file. Use the entries to select new users and folder file names. When the correct settings for the filter appear in the Filter entry field, to edit the selected entry and store it in memory, select **OK**. The action does not alter the filter file on the disk. When you select Save or Save As, the filter file is modified only and stored on the disk.

- **Add**: Adds new users and folders for processing. When you select the Add option, then select OK, the changes are stored in memory only. When you make changes, to store the addition to the disk, select **Save** or **Save As**.

- **Delete**: Deletes an entry in the IncomingFolders Filter list. You cannot leave the list empty. When you select Delete and OK, the changes are stored in memory only. When you make changes, to store the deletion to the disk, select **Save** or **Save As**.

- **Save**: Saves all the modified entries, not just the selected information, into the existing file on the disk.

- **Save As**: Saves all the entries into a new Filter file.

- **Reset**: Resets the values of the selected entries to the original state, not the factory defaults.

- **OK/Cancel**: Approves or cancels the changes. Unless you select Save or Save As, the changes are stored in memory only, not to the filter file that is saved on the disk.

    - **OK**: When you make changes to filter entries without saving the changes, a window appears that prompts you to approve all changes, then to confirm overwriting the file with the modified entries.

    - **Yes**: Stores the changes into the file on the disk.

    - **No**: Aborts all changes from memory and leaves the filter file unchanged.

**AutoRun Filters**

Selecting the **AutoRun** option produces a window where you can select and edit the AutoRun Filters, which are used to determine which jobs are moved automatically from the eligible job queue to the pending jobs queue. When the correct settings for the filter appear in the Filter entry field, to edit the selected entry and store it in memory, select **OK**. The filter file on the disk is not altered and is modified only and stored on the disk when you select Save or Save As.

The following options are available on the IncomingFolders Filter File window:

**Filter Filename**

The Filter Filename menu contains a list of all defined AutoRun filters. Highlight the file name that you want to edit. The entries in the highlighted filter file appear in the lower dialog.

**Set Active Filter**

Sets the highlighted filter to active status.

**Delete File**

Deletes the selected filter file from the list and from the disk.

**Edit**

To produce a dialog that lists user names and incoming folders, select a filter file name, then select **Edit**. Use the entries to select new users and/or folder file names. When the correct settings for the filter appear in the Filter entry field, to save the setting to memory without altering the filter file, select **OK**.

**Add**

Adds new users and folders for processing. Adding information and selecting **OK** stores the change in memory only. After you make changes, to store the addition to disk, select **Save** or **Save As** .

**Delete**

Deletes an entry in the Incoming Folder filter list. You cannot leave the list empty. Deleting information and

selecting **OK** stores the change in memory only. After you make changes, to store the deletion to disk, select **Save** or **Save As**.

**Save**

Saves all the modified entries not just the selected information, into the existing file on the disk.

**Save As**

Saves all the entries into a new Filter file.

**Reset**

Resets the values of the selected entries to the original state, not the factory defaults.

**OK/Cancel**

Approves or cancels the changes.

- **OK**: When you make changes to filter entries without saving the entries, a window appears that prompts you to approve all changes and confirm overwriting the file with the modified entries.

- **Yes**: Stores the changes into the file on the disk.

- **No**: Aborts all changes from memory and leaves the filter file unchanged.


**Dispatch Rules**

To edit the Dispatch Rule File, from the Edit menu, select **Rule Filename**.

The primary Dispatch Rule File window contains a menu where you can select any of the Rule files found in the VIeC Dispatch rules directory. When a Rule file name is highlighted, the lower window displays a list of all the entries found for the selected Rule file.

To delete the selected Rule file from the disk, select **Delete File**.

These following options are available from the Entries in Dispatch Rule File window:

- **Edit**: When you select a Rule file name, a key name is highlighted. To produce a dialog where you can edit the saved values for the selected key name, click **Edit**. Each dialog varies according to the type of key name, for example, string, boolean, and value. The dialogs contain a description of the key name being edited, helpful information such as invalid characters for that field, and a description of how VIeC Dispatch works when settings are changed.

- **Clear**: This option clears all selected entries.

- **Default**: This option sets the values of the selected entries to the factory default settings.

- **Save**: This option saves all the modified entries into the existing file on disk.

- **Save As**: This option saves all the entries into a new Dispatch Rule File.

- **Reset**: This option resets the values of the selected entries to the original state, not the factory default settings.

- **OK/Cancel**: This option approves or cancels the changes.

–   **OK**: When you make changes to rule entries without saving them, a window appears, prompting you to approve all changes and to confirm overwriting the file with the modified entries. To approve all changes and confirm that you want to overwrite the file with the modified entries, select **OK**.

–   **Yes**: To store the changes into the file on disk, select **Yes**.

–   **No**: To abort all changes from memory and leave the Rule File unchanged, select **No**.

## TRACE MENU

Trace menu options allow you to see more detailed information about the job selection process. The Trace menu can be especially useful to determine why a job is ineligible for processing.

The Trace menu has three options:

**Clear Display**

Clears the log console window of all current messages. The Clear Display option does not delete messages placed in the log file.

**Job Selection Loop**

Displays trace messages generated within the scanning portion of the job selection process where jobs are searched to be picked up for processing.

**Job Selection Trace**

Displays trace messages generated during the job eligibility portion of the job selection process where each job is reconciled with every existing Dispatch rule to determine if the file is eligible for processing.

# Using VIeC Web

This chapter contains:

VIeC Web allows you to use the web browser for the following tasks:

• Submit VIPP® jobs

• Create new or additional VIeC user accounts

• Administer VIeC Servers remotely

• Administer VIeC Web passwords and other security functions

Instructions are based on sessions using Microsoft Edge (Chromium). If you use another browser, adjust the instructions if needed.

To change default port settings used for VIeC servers, refer to Changing the Port for HTTP and HTTPS.

## Accessing VIeC Using the Web Browser

To begin using VIeC Web to administer the VIeC jobs:

1. Open a Web browser.

2. In the address field, enter the IP address or URL of the Web-enabled VIeC Server.

3. Click **Go**.

   When VIeC Web finds the server, an **Enter Site** option appears. When you click this option, you can access the Job Status and Submission and Administration tabs.

The Welcome to VI eCompose Server page provides the following options:

| | |
|---|---|
| Job Status and Submission | Provides access to VIeC jobs and functions: |
| | • Submit job |
| | • Delete job |
| | • View job |
| Administration | Provides access to these administrative tasks: |
| | • User and Group Administration |
| | • Server Administration |
| | • Cluster Administration |
| | • Basic Authentication Quick Setup |
| | • Disable Basic Authentication |
| User Icon | When you select the user name or user icon, a pop-up window appears with a **change password** option and a **Log Out** option. |
| | If the user accessed the site with Basic Authentication set to **OFF**, no user name or user icon is shown. |
| Help Icons | When you select the **Help** icon, you can access the online help and an **About** link. |

## Job Status and Submission

The Job Status and Submission page provides access to VIeC job submission functions. From this page, you can access the Job Status Panel and Job Submission panel.

### JOB STATUS PANEL

**User Name**

Allows you to select a user name from the list to view the status of the submitted jobs. If you have administrative privileges, all user names appear in the list and you can view or delete jobs for any user.

**Output Folder**

Allows you to choose one of the available directories to view the status of the jobs submitted to the selected directory.

**Submit Job**

Opens the Job Submission Panel.

The following information is provided for each submitted job:

**Job Name**

The name of the job you submitted. When the job is complete, to display the job files generated through VIeC Web, click the job name link for the required job. If you click a displayed file link, the file is displayed on the system. The way the file is displayed depends on how the system is configured to handle the various file types.

**Status**

The status of the submitted job appears in this part of the table. If the job fails or did not process properly, this entry contains an explanation of the failure, including information about errors and how VIeC handled the job.

**Pages Distilled**

The total number of pages created in PDF format.

**Pages Split**

The number of pages split in the job. A split file can have more than one page within it. For example, an original PDF file that contains three bookmarks is split into three PDF files. If each split PDF file consists of four pages, the total number of split pages is 12.

**File Generated**

The number of files that the job generates.

**Action**

Access job viewing and job deletion using the table in-line controls. To view the job, click the **View** icon. You can view job in the PDF Viewer App. The view list in the Action column is scalable to fit longer lists of files, depending on available screen size, up to three columns wide. Also, a scroll bar is added to present more file links.
You can cancel a pending job or delete a completed job using the option shown in the Action column. To delete the entire job, including the output and the folder, click the **Delete** icon.

## SUBMIT JOB: JOB SUBMISSION PANEL

For descriptions of the files and templates found on this page, refer to Job Creation Area Field Descriptions. Use these options to submit a job to the VIeC:

When you close the panel, it slides in from right to left. Upon closing, the panel slides out from left to right.

**User Name**

Select the appropriate user name for the job. If you have administrative privileges, all user names appear, because you can submit a job for another user.

**Output Folder**

Choose one of the available directories as a repository for the completed job files.

**Job Type**

Click one of these three options for the job type:

- Distill Only

- Distill and Split into Pages

- Split into Pages only

For more information on these options, refer to Job Creation Area Field Descriptions.

**Current Profile**

The Current Profile menu contains a list of the available profiles. Choose the appropriate profile, then to load the selected profile, click **Load**, or to save the profile name entered, click **Save**. The profile file is saved in the `profiles` directory, which is the same directory used by the Job Submission Client. In the **Name** field, enter the file name only. Do not enter path information. For more information on these files, refer to Profile files.

**Output Filename Template**

Select from the list of existing options or create a new Output Filename Template. For more information on creating these templates, refer to Specifying PDF output file names.

**Index File Template**

Select from the list of existing options or create a new Index File Template. For more information on creating these templates, refer to Specifying an index file.

**Data File**

This option allows you to browse for the file to submit to VIeC. When you click **Browse**, the Choose File window opens. Use this window to browse for and select the file or job for submission.
Retrieved files are pulled from VIeC Web as the result of dynamic queries, and are not pulled by direct links to files on the VIeC Server. If the browser provides a choice to either open or save a file, always choose to open the file, rather than save it. After the file opens, you can save it using the **Save** option in the application that the browser used to open the file. If you inadvertently choose **Save**, rather than **Open**, sometimes the webpage that generated the query can be saved, rather than the file itself. When you save files, ensure that the destination path and type are appropriate for the file.

**Initialization (Nub) File**

This option allows you to browse for the appropriate .init file for the job. When you click **Browse**, the Choose File window opens. Use this window to browse for and select the file.

**Cancel**

Closes the Job Submission frame and returns to the Job Status panel.

**Submit**

Submits the job to the VIeC Server.

**Submit and Close**

Submits the job to the VIeC Server and returns to the Job Status panel.

## Administration

The VI eCompose Administration Options page provides access to the following administrative functions:

- User & Group Administration

- Server Administration

- Cluster Administration

- Basic Authentication Quick Setup

- Disable Basic Authentication

For a discussion on the four available security levels supported by VIeC, refer to Defining VIeC Web security levels. Xerox recommends that you set security levels for this application.

### USER AND GROUP ADMINISTRATION

VIeC Web allows the creation of new or additional VIeC user accounts. The default incoming folders produced for each user are:

- alt1

- alt2

- bulk (mandatory)

- default (mandatory)

Additional incoming folders can be added manually for any user on the User Administration page, or the VIeC Web administrator can add an additional line to `C:\ProgramData\Xerox\VIPP\VIPO\vipo.ini` in the following format:

```
vtpw_addtl_user_folders=first_directory_name|second_directory_name|...
```

As an example, after adding this line to the vipo.ini file:

```
vtpw_addtl_user_folders=example1|example2lexample3
```

Adding a new user using the VIeC Web User Administration **Add New User** option results in the creation of these directories for that user:

- example1

- example2

- example3

- bulk

- default

The following information is provided for the user:

**User Name**

Shows the list of all users and identifies the default user.

**Group**

Shows whether the user belongs to the Admin group or the User group.

**Password**

Shows whether the user is the password protected. The password status shows as either Yes or No.

**Action**

The icons in the Action column provide access to the following additional functions: Actions are **Change Password**, **Lock User**, and **Delete User**.

- **Change Password**: To change a user password, click the **change user** icon. Enter the new password in the New Password window, then enter the same password in the Repeat new password window. After you finish entering your password, click the **OK** button.

- **Lock User**: To lock the user, click the **Lock User** icon. At the confirmation dialog, click **OK**.

- **Delete User**: To delete the user, click the **Delete User** icon. At the confirmation dialog, click **OK**.

**Add New User**

To enter a new user:

1. In the right frame of the page, click **Add New User**.

   A menu appears.

2. Enter the new user name in the User Name field.

3. From the **Select Group** field, select the group, either **Admin** or **User**.

4. Enter the new password in the **New Password** field, and reenter the password in the **Repeat New Password** field.

5. Click **Save** or **Save and Close**.

   The new user name appears in list of user names in the left frame.

When a new user is created, the new user does not belong to a group and must be added explicitly to a group. For the user to be able to submit jobs, add them to the user group. To be able to perform administrative functions as an Administrator, add them to the admin group.

**Assigning Users to Groups**

The User Administration feature allows you to classify each user into either one of the defined user groups or admin group. Users associated with the admin group are able to access and complete the administrative tasks found on the VI eCompose Administration Options page. Users associated with the user group can submit jobs to the VIeC Servers accessible from VIeC Web.

To remove the user from a specified group, select the user, highlight the group you want the user removed from, and click **Remove This User From Group**.

### SERVER ADMINISTRATION

The Server Administration feature of VIeC Web allows you to set preferences for the Servers that are part of the cluster of servers accessible from the VIeC Server. The primary Server Administration page contains this information:

**Server Instance**

The names of the servers in the cluster.

Using VIeC Web

## Status

The activity state of each server, which is usually `idle` or `active`.

## Method

The way a job submission order is determined on each server.

## Job

The job that the server is currently processing.

## User

The user assigned to the job that the server is currently processing.

## Distilled

Total number of pages distilled in the job that the server is currently processing.

## Split

Total number of pages split in the job that the server is currently processing.

## Files

Total number of files generated in the job that the server is currently processing.

## Action

The icons in the Action column provide access to the following additional functions:

- **Pause/Resume**: Use this toggle button to pause and resume processing on the server. When you pause the server, if the server is in a busy state, the pause does not occur until the server becomes idle.

- **Halt**: Use this option to cancel the current job and place the server in a paused state.

- **Abort Job**: Use this option to delete the job.

- **View Files**: Use this option to view the accounting file in VTP, .csv, or .xml format.

- **Settings**: There are two functions as follows:

  - **Selection Method**: Use this option to set the job selection method of the server to:

    - First In First Out (FIFO)

    - Shortest Job First (SJF)

    - Longest Job First (LJF)

  - **Exit Server Instance**: This action causes the current job on the server instance to be cancelled, and the process for this server instance is terminated.

    - ⓘ Important: You cannot restart this server instance from your Web browser. To restart a Server Instance, you require physical access to the machine that is running the VI eCompose Server.

In addition to this information, the following options are available for selection:

| | |
|---|---|
| **Refresh Icon** | Use this option to refresh the Server Administration page manually. |
| **Auto Refresh** | Enable **Auto Refresh** to refresh the server administration page automatically. |

To view information specific to a server in the cluster, in the Server Instance field, click the server name link. A new Server Administration page opens.

### CLUSTER ADMINISTRATION

The VIeC Web Cluster Administration window allows you to register the paths of drives that have been mapped or mounted in order to allow access to the root paths of VIeC Servers. VIeC Web displays these additional Cluster Servers on the VIeC Web Application Administration window, where they can be administered in the same way as the primary VIeC Server administered by VIeC Web.

**Register New Server Root**

When you click the **Register New Server Root** button, the system prompts you to enter the path of the new Server Root. Once entered, click **OK** to add the new Server root to the menu.

**Unregister**

You can Unregister or Remove the server root in table in-line control with confirmation dialog.

Refer to VIeC configuration options and use the following discussion to help determine how to set up VIeC Server Clusters:

Assume that you have a two-node cluster set up as follows:
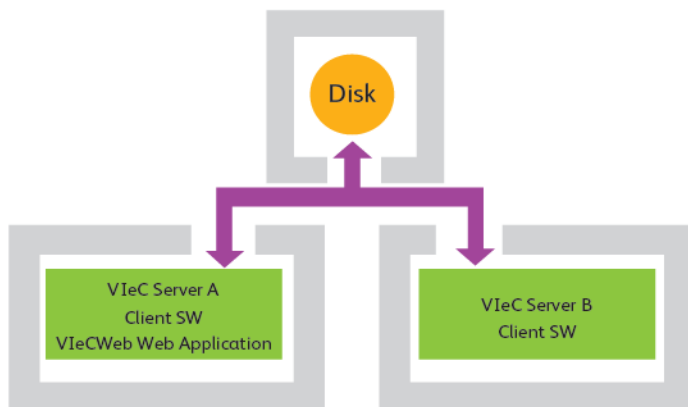


In this configuration, both VIeC Servers service a common job pool resident on a disk, which is shared over a network. Before VIeC Web, administering VIeC Server nodes, such as server pause, server resume, examining logs, and so on, required physical access to the individual VIeC Servers.

With the addition of VIeC Web, VIeC Server A can now be administered using the Web browser:
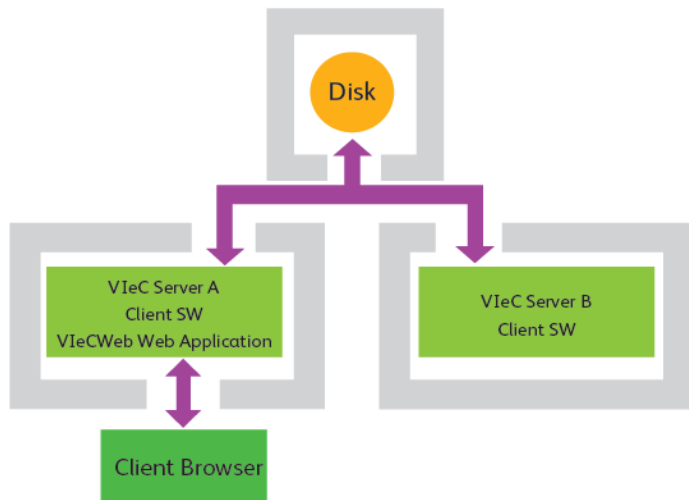
Without Cluster Administration, however, the only way to remotely administer VIeC Server B is to run an additional VIeC Server where:
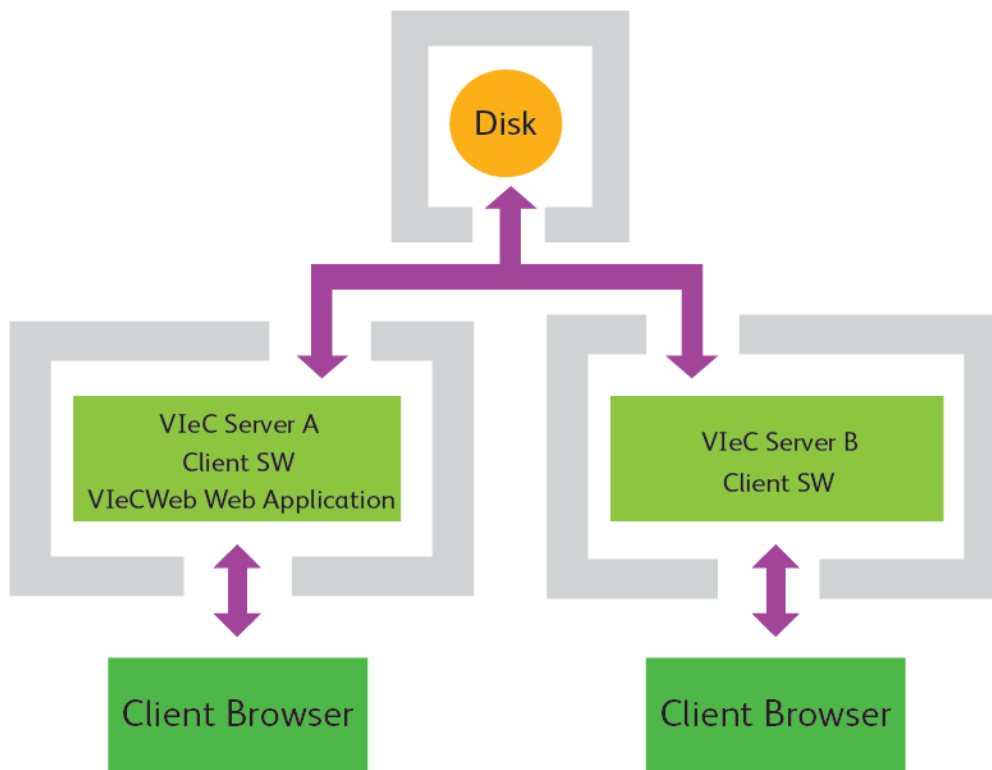


This configuration is inconvenient for several reasons:

- VIeC Web administration with regard to SSH Certificates

- User account passwords

- Multiple access points to monitor with regard to network security, and so on.

The complexity and issues multiply in direct proportion to the number of additional nodes added to the cluster.

Cluster Administration using VIeC Web addresses these issues. To enable Cluster Administration for VIeC Server B, the administrator configures the `X:\Program Files (x86)\Xerox\VIPP\xvtp` directory or the `server root` directory of VIeC Server B to be shared or mounted by VIeC Server A.

Then, from the Cluster Administration window of the VIeC Web web Application hosted by VIeC Server A, VIeC Server B is added to the Cluster Server Root list of the VIeC Web web Application hosted by VIeC Server A, which results in:
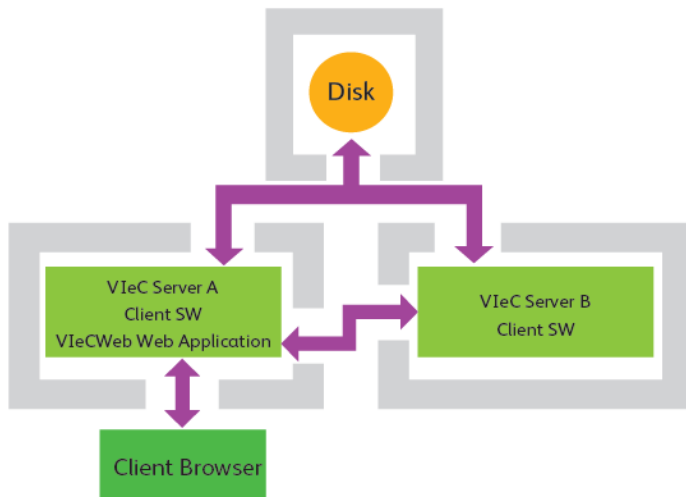


VIeC Server B can be remotely administered over the HTTP connection to the VIeC Web web Application hosted by VIeC Server A, which acts as an intermediary and interacts with VIeC Server B on behalf of the client browser session.

Even if additional nodes are added to the cluster, as long as their server root paths are accessible by the VIeC web Application hosted by VIeC Server A, and added to its cluster server root list, they can all be administered remotely from the single client browser session:
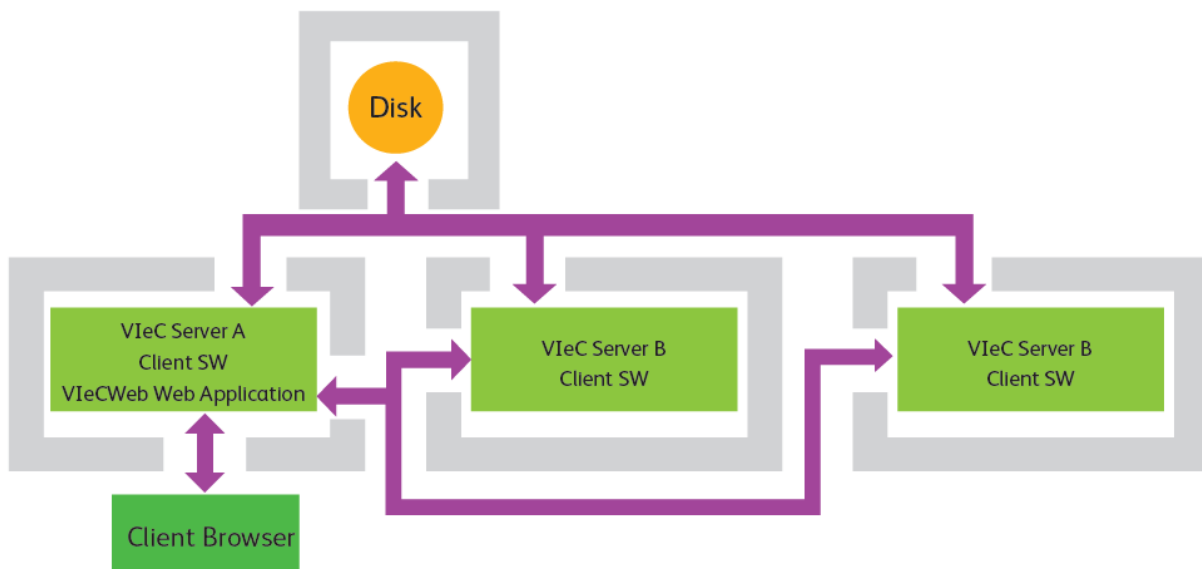
## Basic Authentication Quick Setup

The Basic Authentication Quick Setup process allows user-account level security with user passwords and group security on an area-by-area basis. If the site requires a higher level of security, for more information, refer to Defining VIeC Web security levels.

For a description of the Basic Authentication Quick Setup, refer to HTTP Basic Authentication.

### PERFORMING BASIC AUTHENTICATION

To perform Basic Authentication:

1. Enter the Administrator user name.

2. Enter the password in the **Password** field then reenter the password **Confirm Password** field.

3. Enter the default user password.

4. Click **Perform Quick setup**.
   After Basic Authentication is enabled, you are redirected to the Login page.

### DISABLE BASIC AUTHENTICATION

You can disable the HTTP basic authentication level on the Web Application by choosing **Disable Basic Authentication** on the Administration page, then clicking the **Disable** button. You are redirected to the landing page without the need to log in.

Xerox strongly recommends that you maintain an appropriate level of security on the VIeC Server network.

## Web Browser Notes

When you use Microsoft Edge (Chromium), be aware of these notes:

| | |
|---|---|
| PDF files | If Adobe Acrobat or Reader is installed, when you click a PDF file link, the PDF file opens in Adobe Acrobat or Reader. PDFs encountered as the result of dynamic queries disable the Save As option in Acrobat. To save the PDF file from Adobe Acrobat or Reader:<br>1  Use the Shift-Ctrl-S command.<br>2  Click the small floppy disk icon on the upper left of the IE-embedded Acrobat display. |
| XML files | Unless another tool is associated with XML files, XML files open in a new Microsoft Edge (Chromium) window. |
| CSV files | CSV files open in Notepad, or other tool associated with plain text. |

## Starting and Stopping the Web Application

To start and stop the Web application, follow these steps:

1.  To open IIS Manager, at the Run command prompt, type `inetmgr`, or search for `Internet Information Service` in the Start menu.

    The IIS Manager window opens.

2.  To display all application pools, in the Connections pane, expand the server node and click **Application Pools**.

3.  On the Application Pools page, for the published application that is running, select the **WebAppPool** application pool.

4.  To stop the application pool, click **Stop**. This option stops the normal Web Application or Secure Web Application.

5.  To start the application pool, click **Start**. This option starts the normal Web Application or Secure Web Application.

## Changing the Port for HTTP and HTTPS

1. To open the IIS Manager, at the Run command prompt, type `inetmgr` or search for `Internet Information Service` in the **Start** menu.

   The IIS Manager window opens.

2. In the Connections pane on the left side of the window, to change the port, navigate to the server and default website.

3. In the Actions pane at the right side of the window, click **Bindings Options**.

4. Click **Add**.

5. The Add Site Binding window opens. To set the port, for Type, select **http** or **https**, as needed.

6. Change the port to the needed value. For example, 8000.

7. Click **OK**.

8. If required, repeat steps 5–7 for the other port type. For example, if you selected **http**, you can repeat the steps for https. To enable this binding, ensure that a valid certificate is installed.

9. If you do not want to run or present http, you can remove the default binding port number of 80.

   Now you can access the Web Application from the following URLs:

   - http://localhost:8000/FreeFlowVIeCompose/

   - http://localhost:9000/FreeFlowVIeCompose/

Xerox® FreeFlow® VI eCompose Software User Guide

# VIeC Web Job Submission Service

VI eCompose is a production-level server solution designed to handle a complex set of operations and interactions. Because of this, latency has been built into the system. Although the latency is absorbed when processing large jobs, it can be a limiting factor when processing many small jobs one after the other.

When processing many small jobs, VI eCompose Web Job Submission Service (WJSS) can be a better solution. WJSS is a low-overhead, low-latency, no-frills means of getting a high volume of small, already deployed VIPP® jobs converted to PDF with as little latency between requests and results as is physically possible. In other words, WJSS enables quickly the processing of many VIPP® data files.

WJSS does the processing by pre-instantiating a designated number of Normalizer instances that stay resident, allowing the instances to handle multiple sequential VIPP® to PDF invocations. This action eliminates the per-VIPP®-to- PDF instance invocation overhead present in VIeC. Collectively, these NI (Normalizer Instances) are coordinated and managed by a NIP (Normalizer Instance Pool) with command, control, and status passed between the NI and NIP, usinga fast IPC (Inter Process Communication) mechanism, which eliminates the latency imposed by the disk mailbox and semaphore mechanism used by VIeC.

> Note: There is no support for the standard features offered by VI eCompose such as watched folders, lpr, VPC files, splitting, PDF security, index file creation, accounting, logging, or Dispatch. If you need to use these features, do not use WJSS.

Overall access to WJSS is provided by a .NET proxy Windows Service or as an extension to Microsoft IIS (Internet Information Services). The protocol used is .NET/SOAP.

To take advantage of WJSS, the user is required to develop compatible front-end systems to send Job Lists that contain one or more Job Requests and back-end systems capable of handling and taking action on the messages sent back. Users are responsible to put in place the front ends and back ends to support this operation.

Do not undertake the use of the WJSS client without the support of subject matter experts familiar with your systems, networks, .NET and IIS experience. Manual setup and configuration are required to use this solution. For more information, refer to VIeComposeWJSS.pdf.

# How Xerox® VIPP® Language BOOKMARK Interacts with Dispatch

This chapter contains:

This section explains the relationship between the VIPP® command, BOOKMARK, coded in the VIPP® application, and the Index Template that you create in the VIeC Job Submission client. The relationship is key to understanding how to use Dispatch effectively.

The VIPP® BOOKMARK command creates an interactive bookmark within a PDF file. The BOOKMARK contains information extracted from the VIPP® application, such as name, account number, and email address. The information is used as a key to identify a logical boundary in the application for splitting purposes and as input into the VIeC index file created by the setting in the Index Template. When used with VIeC, the BOOKMARK command can be used to create an extended bookmark rather than a regular bookmark. An extended bookmark is a transparent PDF note, which is usually visible from the comment section of the completed PDF only.

The Index Template is used to define the fields that are contained in the index file, created by a distill and split operation. By default, the index file contains information about the PDF files produced, such as location, name, and so on. By using the Index Template options, users can add additional fields that are extracted from the VIPP® BOOKMARK. This mechanism adds fields such as name, address, account numbers, and email addresses to the index file. Dispatch uses the fields in the index file to verify the job eligibility for dispatch and to extract fields and pass the information to the user-defined process called. As an example, Dispatch can pass the email address to a user-defined process that passes the PDF file and associated information to an email server.

When designing the VIPP® job, think ahead about the dispatch process and requirements. In addition to defining unique fields on which Dispatch triggers, consider what information you want to pass to the process that Dispatch calls. In the example, the information that passes to the Dispatch process includes:

- Personal information such as name and email address

- The title of the email

- The body text of the email that contains variables such as job ID and a time stamp

To begin the process, add a BOOKMARK command to the VIPP® application.

## Set Up the VIPP® Application

In this example, the data for the application, or fields generated by the VIPP® code, such as **SETVAR**, **VSUB**, and so on, contain the following data:

| | |
|---|---|
| Field USER | David |
| Field ACCT | 673987 |
| Field JOBID | 167B |
| Field TIMESTAMP | 20061023 13:35:34 |
| Field VARJobType | Summary |
| Field EMAIL | Dave.Kirk@Xerox.com |
| Field SUBJECT | Please Review |

The target device for this application is the VIeC Server, the intent is to use VIeC to process the VIPP® application, then distill and split the output into individual PDF sets. Dispatch will then be called to pass the required information, and the PDF file, to an email server for emailing to the client. In addition, VIeC will pass the VIPP® data file to an identified VIPP® printer for hard copy printing.

In order to make this happen, information about the client must be collected for the email we will be sending.

Information about the client can be retrieved from the data base fields in the data file and from variables defined in the VIPP® job.

Information for the email typically will be a combination of data base fields and information created in the VIPP® application. For example, the email address is probably available in a data base field, while the text of the email can be created within the VIPP® job.

Below is a sample message created in a VIPP® job. Note that the CS command is used to concatenate several strings. The completed string is left on the PostScript stack, and is called in the next command set using the `$$.` syntax with the VIPP® BOOKMARK command.

```
(Dear $$USER.,\n\nPlease review the enclosed $$VARJogType. and let us)
(know if there are any discrepancies. Call the Helpdesk and reference) CS
(your account number $$ACCT., job $$JOBID., run on $$TIMESTAMP..\n\n) CS
(Sincerely,\nMIS.) CS VSUB
% This is left on the stack
```

Next create the VIPP® **BOOKMARK**.

`($$VARJobType.A$$EMAIL.A$$.) VSUB BOOKMARK`

Note the use of the ^ character between the three variable fields. This has no significant meaning to the **BOOKMARK** command, it is just another character. The character has been placed there to be used as an identifiable field separator in the index template for Dispatch. By adding the field delimiter here, it can be used later to break the three fields apart. You can select any character as a field delimiter, but it cannot be the Dispatch field delimiter `:`, or any other character likely to be used in the bookmark.

The result of this is to generate a VIPP® bookmark that looks something like this when the variables are substituted:

`Summary^Dave.Kirk@Xerox.com^Dear David,`

`Please review the enclosed Summary and let us know if there are any`

`discrepancies. Call the Helpdesk and reference your account number`

`673987, job 167B, run on 20061023 13:35:34.`

`Sincerely,`

`MIS`

In the terminology of the **BOOKMARK**, a single bookmark field was created, which can be split into three sub fields based on the field delimiter ^. The three fields are referred to as fields:

| | |
|---|---|
| F0 | Which is the first field and contains the string Summary and will be used for the email subject field |
| F1 | Which is the second field and contains the data Dave.Kirk@Xerox.com and will be used for the email address |
| F2 | Which is the third field and contains the body of the email text |

## Setting up VIeC and the VIeC Index Template

The Manual Job Submission client is used to set up and configure a job to run in VIeC. You can save the setting entered here can to a profile, which the client you use in production can use.

This discussion is limited to the settings required to run this job. For information about other settings, refer to the *FreeFlow VI eCompose Users Guide*.

In this example, you set the Job Type to **Distill and Split into Pages**. To use this setting, the following requirements are necessary:

- The VIPP® application has a valid BOOKMARK command, to enable logically VIeC to determine set boundaries.

- An Index File Template exists to create the index file.

When you first start the Job Submission client, the default index template is selected. To use Dispatch, customize this index template.

The default index template has eight fields and a field delimiter command. A description of each field is provided later in this document. The default fields are:

- <FS=','>FileName
- InputFile
- EndSplitDate
- EndSplitTime
- OutputDir
- Pages
- FileSequence
- Bookmark

To meet the requirements of the job in this example, you edit the Index Template, and replace the Bookmark field with three new fields. The three new fields are based on the sub fields that were created in the VIPP® bookmark, generated by the **BOOKMARK** command in the VIPP® application example.

The edited Index Template for this example, looks as follows:

```
<FS=','>FileName,InputFile,EndSplitDate,EndSplitTime,OutputDir,Pages,
FileSequence,Subject|BFS='^'[F0],MailTo|[F1],EBody99|[F2]
```

In this index template, the index delimiter and the first seven fields remain unchanged. The Bookmark field is deleted and replaced with three new fields. Field names are user defined. Choose the field names with care, because one or more of these field names can be used in the Dispatch rule. Using unique identifiers helps to ensure that the Dispatch rules match a single job, based on the names in the index file.

The F0 syntax is used to select the data in the first field of the **BOOKMARK** command. In this example, the command contains the string Summary. When passing information to the email server, both the subject and the subject field for the email is passed on. Likewise, the F1 syntax puts the email address in the index field MailTo, and the text that makes up the body of the email is placed in the index field called EBody99. For larger messages, review the extended **BOOKMARK** command, which allows the **BOOKMARK** to exceed 256 characters.

To summarize the discussion to this point:

- VIPP® extracts field contents from data fields and variables during normal execution.

- Ensure that the data Dispatch process and User Defined process required are present in the VIPP® **BOOKMARK**. Both the processes occur inside the VIPP® job during the execution of the VIPP® **BOOKMARK** command.

- The Index Template extracts data from the data populated in the **BOOKMARK**.
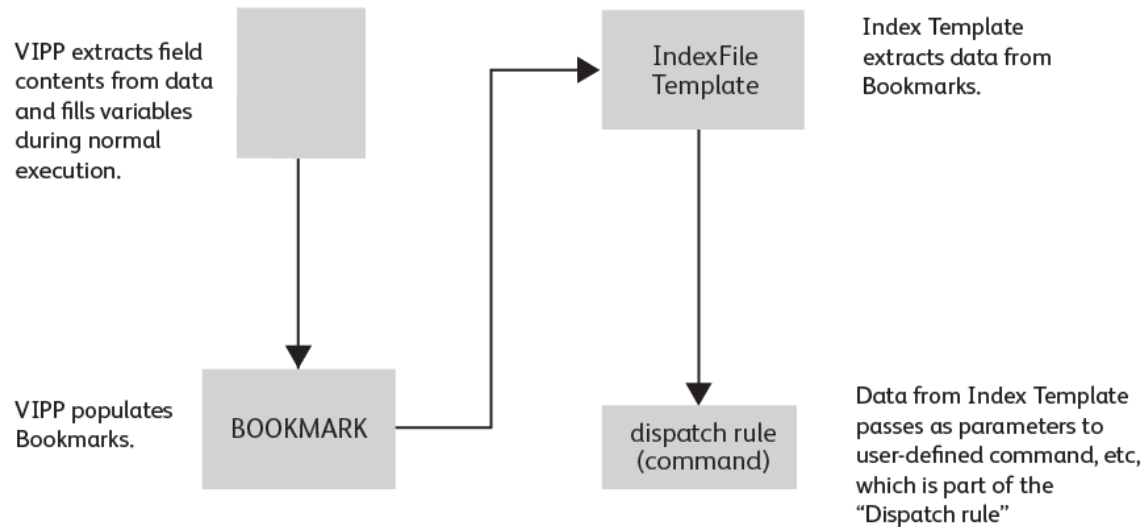
## Passing Data from the Index Template

The next step is to pass data from the index template as parameters to the User Defined process. This process is defined in the Dispatch rule.

After VIeC processes the Job, and after the PDF files are created and deposited into a folder, you can configure VIeC to perform user-defined actions on PDF files automatically. This activity is handled through the Dispatch module. The Dispatch module starts a user-defined process that is actioned against the PDF files that the job produced, such as moving the files to the next step in the workflow.

Dispatch is a module that can execute a user-defined program and pass to it the required arguments and the PDF file. A typical use of Dispatch is to forward the PDF file, and arguments such as email address, subject field, and email body, to an email server.

When activated, Dispatch checks whether a job processed by VIeC is eligible to be dispatched. VIeC first determines eligibility using the Incoming Folders Filter. The Incoming Folders Filter is where users and folders are defined. For VIeC to consider a job eligible for dispatch, an entry in the Incoming Folder Filter dialog box is required. Ensure that this entry matches the user name that sent the job, and the target folder to the defined area where the PDF files for that job will be located. If the criteria is not met, VIeC ignores the job, and does not consider it for dispatch. If the criteria is met, the job is tested to determine whether to run the job automatically or start the job manually.

After VIeC determines that the job is eligible to dispatch, the Dispatch module checks the Autorun filter to determine whether to start the job manually for dispatch, or if the job is set to auto run dispatch. The Autorun filter allows the user to select either a manual or automatic startup of Dispatch. The Autorun filter is similar to the Incoming Folders Filter, in that both the user name and the folders that have the setting Autorun are defined. Dispatch processes jobs automatically that match the entries in the Autorun dialog box. Start all other jobs manually in the Dispatch module.

The next step is to verify the Dispatch rule, because only those jobs that meet the rule criteria are processed. All other jobs have a status that indicates that user intervention is required.

A Dispatch rule determines whether Dispatch can process the job. The Dispatch rule defines the user-defined program or script to call. The called program or script defines the actions taken, such as move the PDF from folder A to folder B, or pass to another sub system, such as an email server. For more information about Dispatch rules, refer to Dispatch Rule Files.