

# Elliott® V8.5 Release Notes

## Introduction

Elliott Version 8.5 is a major release after V8.2. If you do not use the new feature of alphabetic document number, then no conversion is required. On the other hand, if you do want to use the alphabetic document number feature, database conversion is required. Other external applications, like Crystal Reports or web applications, may also need to be converted and may make the conversion process more significant.

The Elliott V8.5 database is highly compatible with Elliott V8.2. The existing table and field sizes are the same. If you don't convert to V8.5 specific database formats, you can use Elliott V8.2 with V8.5 between different desktops and workstations. You can also run Elliott V8.2 and V8.5 on the same desktop, as long as it is not at the same time. This interoperability allows you to evaluate the new features in Elliott V8.5 before you switch everyone to this new version. Therefore, upgrading to Elliott V8.5 is relatively easy with minimum risk as long as data conversion is not needed.

The following is a list of major features in Elliott V8.5:

- Support Alphabetic Document Number
- Run Local
- Licensing with Key Code

For a general description of other changes since Elliott V8.2, please refer to the section “What’s New since Elliott V8.2.”

## ***Support Alphabetic Document Number***

The primary reason of this feature is because some Elliott users are running out of invoice numbers and are approaching invoice number 999999. In the past, users could use a special archive procedure to reset the invoice number. However, there are some limitations with the archive and users may find some minor issues after the archiving.

By using alphabetic invoice numbers, each digit can have the value from 0 – 9 and A-Z. So the maximum number of invoice numbers is  $36 * 36 * 36 * 36 * 36 * 36 = 2,176,782,336$ . This exponentially increases the number of invoices that can be stored in Elliott, so users can avoid resetting the invoice number counter in the future.

The alphabetic document number feature not only applies to invoice numbers, it also applies to most other fields that are considered as document numbers. This includes, but is not limited to, sales order numbers, purchase order numbers, AP voucher numbers, and work order numbers. Each kind of document number is controlled by its corresponding Global Setup flag. You will need to turn that flag on in order to use alphabetic values for that document number.

## ***Run Local***

By design, Elliott ran from a network share in the past. This offered the benefit of being able to install updates in a centralized area without the need to install local client updates on each

workstation. In recent years, out of security concerns and other reasons, Microsoft is making it more difficult to run from network share with the use of artificial road blocks that Elliott needs to jump through.

Starting with Elliott V8.5, users can run Elliott from local folders even though the data still resides on the network share. This not only bypasses the limits placed by the Windows OS when running from network share, it also improves application performance and makes program execution more reliable. You will continue to install Elliott updates on the network share. In the meantime, when executing Elliott V8.5 programs from local, the system will automatically synchronize the updated network programs to the local folder. Therefore, there's no need for Elliott local programs to be manually updated.

### ***Licensing with Key Code***

In the past, Elliott used license files to determine your licensed users, applications and maintenance date. We emailed you, and you manually copied the Elliott license files to your Elliott folder to reflect the new licensing. Starting with Elliott V8.5, all licensing updates are controlled through a 30-digit license key code similar to the software industry standard. The new license key code will be emailed to you, and you can simply copy and paste it to your Elliott configuration area to reflect the new licensing.

## **Requirements**

Elliott V8.5 requires the Pervasive® PSQL 11, 12 or 13 database engine to run. You may obtain a copy of PSQL from your Elliott reseller if you do not already own a copy. For the best performance, we recommend the 64-bit version of PSQL 13.

### ***Client Operation Systems Support***

Like Elliott V8.2, V8.5 no longer supports Windows XP, Vista, Windows Server 2003 and 2008 32-bit and R1. At a minimum, you must have either the Windows 7 or Windows 2008 R2 operating system installed to run Elliott V8.5. But we suggest you upgrade to newer Windows OS because Microsoft will stop supporting Windows 7 and 2008 R2 in January 2020.

Elliott V8.5 will run on the following client operating systems:

- Windows 7 (all versions)
- Windows 8 (all versions except Windows RT)
- Windows 10

### ***Server Operating Systems Support***

Elliott V8.5 will run on the following server operating systems:

- Windows 2008 Server R2 (64-bit only)
- Windows 2012 Server
- Windows 2016 Server
- Windows 2019 Server \*

\* You will need to download the latest PSQL 13 services pack from Actian to support Windows 2019.

Unless you login to your Windows server as “Administrator” when running Elliott directly on a Windows 2016 or 2019 server, you may encounter an NTFS permission issue due to UAC security elevation. This can be a problem even for an admin equivalent user. We recommend you follow the Elliott NTFS Security implementation guideline to address this issue. Please go to <http://support.elliott.com> and find the KB article “Elliott 8 Directory Structure and NTFS Rights” for more details.

Running Elliott V8.5 directly on most of the modern Windows servers requires adding Elliott EXE files to the DEP (Data Execution Prevention) list. Starting with Windows 2016, the DEP list no longer supports defining EXE on a mapped drive, which is needed for Elliott. In that case, you have three options:

1. In DEP setup, choose “turn on DEP for essential Windows programs and services only.” In essence, this turns off the DEP checking for application programs. This is the simplest solution to address DEP issues if you intend to use the server for Elliott Database engine purposes only. But it may present potential security issues for a remote desktop server if you use other applications like Internet browsers or emails.
2. Use Run Local Mode feature. If you do not want to turn off DEP for security considerations, then this is your best solution. Your users can use the provided EL850.EXE utility to create the DEP list automatically when creating the desktop Run Local icon.
3. Use Run Hybrid Mode feature. This is like Run Local, but all users on the server share the same local copy of Elliott. Therefore, you only need to define the DEP list one time. If you have DEP issues that can’t be resolved with the Run Local mode, consider using the Run Hybrid mode instead.

### **Microsoft®.NET Framework V4.5**

This version of Elliott requires Microsoft.NET Framework V4.5. If it is not already installed on your computer, you may download it from [www.microsoft.com](http://www.microsoft.com) . All new PCs come preinstalled with Microsoft.NET Framework V4.5 or V4.6.

### **Known Limitation**

General speaking, you should avoid exporting to or importing from a file name or its path if it contains spaces. But export processor and mass email programs are special cases. The export processor and mass email programs do support exporting to file and path with spaces.

## **Areas Requiring Attention When Upgrading to V8.5**

### **Convert Database**

Since Elliott’s document numbers were indexed as numeric in the past, if you wish to start using the Alphabetic Document Number feature, you will have to go through a conversion process to re-index these columns as strings. Once you actually store alphabetic document numbers in these fields, there will be no reversing back to the Elliott V8.2 database format. It is unlikely that you will need to go back to the Elliott V8.2 database format due to any Elliott V8.5 issues. But you may use other third party applications -- like Crystal Reports or web applications -- that access



## Step 2 –Convert Database, Test or Live

After you are done with your conversion, to begin using the V8.5 database, like ELI85DATA, you will need to convert your V8.2 data to Elliott V8.5 format. This involves rebuilding the index of many Elliott tables. For example, Invoice Number used to be a numeric field. It is a string field in Elliott V8.5. For all tables that have the invoice number field and are part of the table's index, you will need to convert those indexes to treat invoice numbers as string fields now. You can do so by logging on to the PSQL server as an administrator, bringing up the command prompt and going to the folder <ElliottRoot>\Bin85, and typing the following command:

```
DDF2BTR <ElliottRoot>\DATA\*.BTR
```

Substitute <ElliottRoot> with the root directory where your Elliott is installed. For example, use "M:\Elliott7". The DATA is the corresponding DATA folder, it can be DATA, DATA\_02...etc. which corresponds to each company. If you have a custom modification with your database, you will specify the path of your custom DDF with the DDF2BTR command. Use DDF2BTR /? to find out the proper command to do so.

When you perform this conversion on the PSQL server, you can estimate about 100 MB per minute. Do not perform this conversion on the client side for the performance will be significantly slower and less reliable. The actual time may depend on many factors. Only convert your Elliott data when no one is using Elliott, including external applications that access Elliott data like Report Writer or web services.

To be safe, you can optionally copy your production company's data to a new company and perform your database conversion in the new company first, then continue your test with external applications that point to this new company. If the external application tests are successful, then convert your production company data. If you have multiple companies, you may perform this conversion one at a time. You don't need to convert all of them at the same time. You still can convert the V8.5 database back to the V8.2 format as long as there's no alphabetic value stored in any document number fields.

## Step 3 –Finally Start Using Alphabetic Document Numbers

This step is optional. But if you do need to use alphabetic document numbers, you will go to Global Setup -> System -> Comp. Specialized Control. Answer "Y" to "Use Alpha Document Numbers?" Then you will indicate the document for which you intend to use alphabetic values. For example, you may choose to use an alphabetic value for an invoice or sales order number, but not for a purchase order number. Once you are done with this, you will go to the corresponding counter area to set the starting value. For example, if you choose to use alphabetic invoice numbers, you may go to A/R Setup and set the starting invoice number like, say, AAA001. This value is up to you. The important thing to know is if the digit in the starting value is alphabetic, then it will stay alphabetic. If it is numeric, then it will stay numeric. So the digit of value "A" can have a value of A-Z. The numeric digit will have the value of 0-9. Therefore, this format can accommodate a maximum  $26 * 26 * 26 * 10 * 10 * 10 = 17,576,000$  number of invoices. It is up to you to decide your invoice number format. Alphabetic values can be in the beginning, the middle or the end. Or you can make them all alphabetic. The format will stay the same until you change the starting counter value. Once you have alphabetic values stored in any document number fields, you will not be able to go back to Elliott V8.2

## ***Set Up to Run Local***

By Default, Elliott V8.5 runs from a local folder even though Elliott V8.5 is installed in the network share. To setup a workstation to run local, you will use Windows Explorer to browse to the folder <ElliottRoot>\Bin85 and run on the EL850.EXE application. This application will ask you which Elliott application to create. By default, Elliott V8.5 (EL850CC.EXE) will be checked. Accept the default and click “Create” to create the icon and Windows Program Menu. Before creating the icon on your desktop, this program will check if the Elliott application programs are already setup in your local folder. If not, it will copy the necessary files from the network to your local folder. Your Elliott programs are copied to your local user profile folder as follows:

C:\users\\appdata\local\Netcellent\M\Elliott7\bin85

“M” is the network drive where your Elliott V8.5 is installed. “Elliott7” is the folder where your Elliott is installed on the network drive. The startup executable for Elliott V8.5 control center is EL850CC.EXE. Since this path is long, in the future, when you try to run Elliott from your local, simply click on the Elliott V8.5 icon on your desktop to run.

If you are running Elliott on the server, you should select the “Create DEP Exclusions (for Running on Server)” option within the EL850.EXE application. If you do not, it is likely you will receive Error 114 when you start up Elliott. Please refer to the section “Fix DEP Issue If You Get Error 114 When Starting Up Elliott” below for more details on this issue.

Each Run Local program folder is about 310MB in size. Imagine if you have 20 users on a terminal server, and for each user you create a Run Local folder in the user profile. That can take up to  $310\text{MB} * 20 = 6.2\text{GB}$  of disk space on your C: drive (the default volume where your user profiles reside.) If this additional disk space consumption is not an issue, then Run Local should be the best solution for you because it has minimal potential for software update conflicts.

One of the biggest advantages of running Elliott 8.5 from a local folder is that you can update Elliott 8.5 network folder’s programs anytime during the day. The next time the run local Elliott user starts up Elliott 8.5, their local copy will be synchronized automatically. This brings up an important point. Some of your automated processes like deferred processing tend to stay logged in for a long time. To cause the update of the local programs used by defer processing, you need to close and exit the deferred processing programs and start it up again after your Elliott 8.5 update has been installed. Otherwise, you will continue running on the older version of Elliott 8.5 with your deferred processing.

## ***Set Up to Run in Hybrid Mode***

Hybrid mode can be used if you run Elliott on the terminal server and you have disk space concerns with the Run Local option. But if you can afford the additional disk space, we highly recommend you to use “Run Local” option to avoid software update conflicts. In some rare situations, if you have difficulties in resolving DEP issues with Run Local, then you can consider using the Run Hybrid mode. Elliott Hybrid mode is similar to Local mode, where the Elliott programs are copied to the local folder. But in Local mode, each user will have their own copy of Elliott programs in their profile folder. With Hybrid mode, all users on the server will share the same programs in a central directory. For example, you may setup the following folder on your server:

D:\Netcellent\M\Elliott7\Bin85

The easiest way to setup Hybrid mode is to use the EL850.EXE application. By default, the Local Folder will be similar to the following:

C:\users\\appdata\local\Netcellent\M\Elliott7\bin85.

Click on the button “Change” to choose the drive or a folder. System will automatically append a path similar to \Netcellent\M\Elliott7\Bin85 to the drive and folder you choose. The key thing to remember about the Hybrid mode path is that it must have the folder name “Netcellent.” After that comes the drive letter, the Elliott folder name, and Bin85. You can have whatever you’d like preceding the “Netcellent” part. So in the case of:

D:\Netcellent\M\Elliott7\Bin85

it implies the corresponding Elliott V8.5 folder on the network is at “M:\Elliott7\Bin85” and the local hybrid folder will synchronize with that network folder. The icon for Hybrid mode will be setup for you by the EL850.EXE application. If this is a terminal server, it is a good idea to choose the option of creating a shortcut for “Everyone on this machine” so you don’t need to create Elliott shortcuts for other users on the same terminal server. If you do not want to do that, you can manually create the icon on each user’s desktop by using the following sample values:

Target: D:\Netcellent\M\Elliott7\Bin85\EL850CC.EXE

Start in: D:\Netcellent\M\Elliott7\Bin85

When you run Elliott on most of the modern servers, you are likely to receive error 114 due to DEP issues. To resolve a DEP problem, you should make sure the check box “Create DEP Exclusions (for Running on Server)” is selected. If you did not do that, you will define the Elliott executable in the DEP exclusion list. Please refer to the section “Fix DEP Issue If You Get Error 114 When Starting Up Elliott” for more details on the DEP issue.

Since all users running in the Hybrid mode can cause updates from the network to local, you should not implement NTFS security to restrict your Elliott users from updating the local Elliott folder. Alternatively, the system administrator should take on the responsibility to immediately go to each server running in the Hybrid mode to start up Elliott and initiate the synchronization. With Hybrid mode, you should avoid updating Elliott during the day since the next user that brings up Elliott in Hybrid mode will initiate the synchronization process and may run into an update conflict.

### ***Continue to Run in Network Mode***

Running Elliott in Network mode is the traditional method and is still supported, but not recommended, in Elliott V8.5. You will have to setup the Elliott icon manually yourself. For example, if your Elliott V8.5 resides in network folder “M:\Elliott7\Bin85”, then you will setup the icon with the following sample value:

Target: M:\Elliott7\Bin85\EL850CC.EXE

Start in: M:\Elliott7\Bin85

### ***Fix DEP Issue If You Get Error 114 When Starting Up Elliott***

When you start up Elliott and immediately receive an Error 114 message, then you have a DEP issue. If you are running Elliott on a workstation, chances are you won’t encounter DEP issues. On the other hand, if you are running Elliott on the server and DEP is not turned off, then you

should add DEP Exclusions for Elliott's EXE. Otherwise, you are likely to receive Error 114 when you start up Elliott.

With Elliott 8.5, you can add DEP Exclusions easily with the EL850.EXE utility when you use it to create Elliott Run Local or Hybrid folders and desktop icons. The "Create DEP Exclusions" option will be checked by default. As a result, you don't need to manually define them as in previous Elliott releases. Under most situations, this utility will solve your DEP issue. But in some situations, if the DEP problem still persists, we suggest you use the following solutions:

- (1) Add DEP entries in the Windows DEP Tab. The EL850.EXE can add the DEP Exclusions by adding to the registry the HKLM area, which mimics the Windows DEP tab behavior. To do so requires admin rights. If your user does not have admin rights, then you can add the DEP entries for that user manually through your login. See the section entitled "Add DEP Entries in Windows DEP Tab" below.
- (2) In some situations, we have noticed that the DEP entries need to be added to the Registry HKCU area when the user needs to run Elliott from the user profile folder. The EL850.EXE will do so automatically. But if the process fails for security or other reasons, then the user can add to the HKCU registry manually. Look for the KB article entitled "Manage Data Execution Prevention (DEP) Through Registry Editing" on <http://support.elliott.com> for more details.
- (3) If both (1) and (2) fail, implement the Run Hybrid method above. Look for the KB article entitled "The Run Local or Hybrid Feature in Elliott 8.5" on <http://support.elliott.com> for more details.

### **Add DEP Entries in Windows DEP Tab**

Bring up Computer Properties and go to "Advanced System Settings." Under "Performance," click on the "Settings" button. Go to the "Data Execution Prevention" tab. If you are running Elliott on the server only for the admin user or Defer Processing, we suggest that you select "Turn on DEP for essential Windows programs and services only." This will require rebooting your server in order for this change to take effect.

If this is a terminal server, we suggest that you define Elliott V8.5 programs in the exclusion list. Depending on whether you are running in Local, Hybrid or Network mode, browse to the corresponding Elliott programs folder and exclude the following Elliott V8.5 EXE programs from DEP:

DDF2BTR.EXE  
EL850CC.EXE  
EL850DP.EXE  
EL850EV.EXE  
EL850GE.EXE  
EL850ME.EXE  
EL850P.EXE  
EL850RP.EXE  
EL850TK.EXE  
EL850US.EXE

You may search using keyword DEP or 114 on <http://support.elliott.com> for more details.

## ***Defer Processing***

If you start using alphabetic document numbers, certain deferred processing tasks setup previously in Elliott 8.2 may not work correctly after converting to Elliott 8.5. These are the deferred processing tasks that prompt for starting and ending document numbers. For example, the COP Invoice Posting task prompts for a starting invoice and an ending invoice number. When you setup deferred processing for this task in Elliott 8.2, you most likely hit the Enter key on the “Starting Invoice Number” field, which causes the system to default the starting invoice number to 000000 and ending invoice number to 999999. After you start using alphabetic invoice numbers, this task that was setup in 8.2 will not post any alphabetic invoice numbers. The solution is to delete this deferred processing task and set it up again in Elliott 8.5. Review your deferred processing tasks and decide which ones need to be setup again in Elliott 8.5 after you start using alphabetic document numbers.

## ***EL850U.CFG File***

If you have the file EL800U.CFG in your existing <ElliottRoot>\Bin folder, then you should copy this file to the <ElliottRoot>\Bin85 folder and rename it to EL850U.CFG. This file contains override information pertaining to EL850S.CFG. The installation utility does not copy or create the EL850U.CFG file for you. Most people do not use this file. But if you use it, you will need to handle this step manually.

## ***Add Read/Execute NTFS Right to Bin85 Folder***

When you install Elliott 8.5 on top of an existing Elliott folder, a Bin85 folder is created to store your Elliott 8.5 programs. If you previously implemented NTFS security on your current Elliott folder, you should assign NTFS security to the new Bin85 folder to match what you did to the Bin folder. Typically, you will give the user group “Everyone” the “Read and Execute” rights to the Bin85 folder.

## **What’s New since Elliott V8.2**

Below is a list of new features that have been implemented since the Elliott V8.2 general release. Detailed information about these features can be found online at <http://support.elliott.com>. Please search there to locate the appropriate documentation if you wish to learn more. If you are upgrading from an earlier Elliott version, you may find the “What’s New since Elliott V8.1, V8.0, V7.5, V7.4, V7.3, V7.2, V7.1 and V7.0” sections informative.

## ***Elliott Support Portal at <http://support.elliott.com>***

You may find out the details of all features discussed in this section on <http://support.elliott.com> by doing a keyword search. This is Elliott’s support portal, where you can search for documents and Knowledge Base articles. You may also suggest new ideas of how Elliott can improve using this portal. All new features since 2015 have been documented here. We encourage you to visit this website as your first stop to find more information about Elliott Business Software.

## ***Support Elliott Payment Portal***

Elliott supports ACH payment processing in the AR module -- a change that was made since V8.2. We now have an Elliott payment portal where your customers can pay you online through ACH. For example, you may pay your invoices from Netcellent at

<https://paymyinvoices.us/netcellent>

You may not yet be authorized to access this site. In that case, you will need to make a request to Netcellent first. The important thing is you can evaluate how you can pay Netcellent through ACH, and determine if you wish to have the same feature yourself so your customers can pay you online with ACH.

To support the Elliott Payment Portal, we added a flag to eContact to determine if an eContact can access the payment portal online. We also offer the option to eliminate empty cash receipt posting journals. This way, the user can schedule deferred processing to automatically post cash receipts every few minutes for payments received through the payment portal without creating multiple empty cash receipt journals.

## ***Improved PDF PostOffice***

PDF PostOffice was introduced in Elliott V8.2 to email customers using professional-looking PDF invoices, statements, order/RMA acknowledgements, quotes, or vendors for POs. This eliminates the need to print hard copy documents, and saves manual labor, materials and postage. Since V8.2, we have also made improvements in the following areas: (1) PDF PostOffice documents can now be sent to salesmen based on the eContact-defined for the salesman; (2) PDF PostOffice documents can now be optionally sent to the bill-to customer of the sales order; (3) To help automate PDF PostOffice Order Acknowledgements being sent to the customers, we can optionally eliminate the “Nothing in Range” message when printing order acknowledgement; (4) To help users setup the recipients for PDF PostOffice documents, the corresponding flags of PDF PostOffice flags in eContact can now be updated through CSV import.

## ***Mass Email & Export Processor Record Selection***

In addition to the “Filter” tab in Mass Email & Export Processor, you can now select the records based on an ASCII file which contains the primary key of the selected table. For example, if you are processing customer records, then provide a file with one customer per row and each row begin with the customer number. This option is also supported through the command prompt.

## ***Links Improvement***

In the past, we displayed the date when the user links were added. Since more people are using “Automatic Programming Link” now, the date the link type was added which is not very helpful. With this update, we now display the actual document date in the link list window. In addition, you can now import links through a CSV file. The use case for this is to support drill down to the original invoice document in A/P summary account inquiry, if you don’t have control on the naming convention of the scanned document.

## ***Events Improvements***

By design, a trigger event will send emails to subscribers without the need for approval of the users who trigger the event. The idea is that the subscriber’s need to know certain events can’t be

tampered with. But in Elliott V8.5, we now introduce a flag in User Setup to determine if we should prompt a user when the event is triggered. Typically, this would be a supervisor-level user where we give that user the discretion on whether to send the event emails or not.

In addition, we have added the following new events: (1) Change Order's Salesman; (2) Enter Customer in Maintenance or Inquiry; (3) Held Order Released; (4) Shipment Data Update; (4) Add New Purchase Order Header (before printing of PO); (5) Add New Purchase Order Line Item (before printing of PO); (6) PDF PostOffice Email Sending; (7) Print Duplicate Invoice from Invoice History.

### ***Improved SMTP Email Support***

SMTP is the default method for sending emails since Elliott V8.0. In early 2018, Netcellent switched from an internal Exchange server to Microsoft Outlook 365 (hosted email servers). As a result, we discovered various issues related to working with email hosting providers. The following are improved SMTP features in Elliott V8.5: (1) Generally speaking, your email hosting providers do not want you to perform mass emails for fear of spamming, so they place limits on you. In the past, if you ran into the limit established by your provider, you simply got an error in the email log. Elliott now has flags to allow throttling of sending emails so you can avoid overrunning the limit; (2) For mass emailing, you will need to work with special hosting providers. These special mass email hosting providers may require long server URLs, long user IDs, or long passwords. Elliott has been significantly expanded to allow setup of longer values for these fields; (3) When sending emails to hosting providers through the Internet, we discovered that you may receive maybe one connection error per 100. This is the nature of the Internet. We have improved the re-try logic to eliminate this type of error.

### ***Active Users Report***

Under Password Setup, List, there's an option to print the "Active Users Report". These are the users currently logged into Elliott. You can use it instead of User List window. In addition, you can defer this report to run at night time to find out which user did not log out of Elliott at the end of the work day (which can be a potential problem of locking up records or files). You can optionally send an email to the logged in users to remind them to logout at night time.

### ***New Export Processor Features***

Export Processor has been enhanced to do the following: (1) Support exporting 12 months of history in various tables, including, but not limited to, customers, customer types, salesmen, inventory locations, vendors, ship-tos, ship-vias, term codes, and tax codes; (2) Many additional tables are supported, including files in GL, AR, IM, PO and BOMP modules; (3) We now have the option to export CSV files with the "=" sign to avoid Excel dropping the leading zeroes of the column value.

### ***Phantom Locking***

In the past, when a user brought up an item in Change mode, that item was locked. While that was happening, if another user tried to update the same item (e.g., placing order for the item) that user would receive the "Item is locked by..." message until the first user finished the change and unlocked that item. This type of conflict tends to increase with large Elliott user sites. In the

past, we tried to educate our users to finish changing an item or customer quickly to avoid locking conflicts.

In V8.5, we have made changes to reduce the item and customer locking conflicts. For example, if a user brings up an item in Change mode, the item record is not locked. Therefore, other users still can update the item record. When the first user finishes changing that item, before the changed values are written back, the system will check if that record has been changed in between. If not, the item changes will be written back. On the other hand, if that item record has been changed in between, the first user will receive a message indicating the change was not successful because other users have made changes in between, and it will ask the first user to try again. This method is called “phantom locking” and significantly reduces the locking conflicts. Because most of the locking conflicts in Elliott are related to item and customer file maintenance, so we only implemented phantom locking in these two tables for now.

### ***Improve Point of Sales (POS) Credit Card Processing***

We now support Payware POINT interface which will support chip cards and near field communication (NFC) type of cards which include Apple & Google pay. We also support customizing the comment section at the bottom of POS receipt.

### ***More Reports That Can Optionally Export to CSV Files***

The following are more reports can now be exported to CSV files:

- (1) PO Edit List – In additional, we support both the summary and detail versions of the CSV file. The summary is one PO per line. The detail is one PO line item per line. The detail format also matches the PO Line Item Change CSV import. The idea is you can export a PO line item CSV out through the PO edit list. Edit the data in Excel and import the change back to update the PO.
- (2) Inventory Trx Audit Trail Report
- (3) Item Audit Trail Report
- (4) Physical Count Tag Pre-Post Update Edit List
- (5) Count Tag by Item/Warehouse Variance Report
- (6) BOMP End Item Where-Used Report
- (7) BOMP Costed Bill of Materials Report
- (8) BOMP Indented Bill of Materials Report

### ***Distribution History***

In Elliott, we have AR, AP, PR, IM and BM distribution files. The content in the distribution files will be interfaced to the GL module and purged. But in reality, many Elliott users do not purge the distribution after interfacing because the distribution file has more detail information than the General Ledger transactions. The consequence of not purging the interfaced distribution records is that the interfacing is becoming slower and slower over time.

In Elliott V8.5, when the distribution records are purged, they are now moved to the distribution history. Users can print distribution history just like the distribution file. Therefore, there is no reason not to purge the distribution after interfacing to GL. As a result, the distribution files will become smaller and the GL interface can be significantly faster.

## ***Print/Purge System Activity Log***

The System Activity Log stores users' login and logout information from Elliott and web services. It also stores credit card processing and email sending activities from Elliott. This file tends to grow big over time and takes up a significant amount of disk space. We are now providing a utility for users to print data from this file and offer users the option to purge this file to trim down the disk space required.

## ***Recalculate Order Total Amount with Order Integrity Check***

When you enter a COP sales order, the order total amount may or may not be updated. It depends on whether you access the order billing screen or not. In the same manner, when you change the order, the order total amount may not be updated depends whether you access the order billing screen. Therefore, you can't rely on the order total amount in CPORDHDR (COP Order Header) table when you create Crystal Reports or with other third party applications. Our common advice is that you need to calculate the value from COP Line Items table (CPORDLIN.) But that can be complicated, especially you need to consider the possibility of partially post, unit of measure, discount percent...etc.

Currently, Elliott guarantees the order total amount will be re-calculated after you print the pick ticket or perform billing selection. We now offer a third option, when you go to Global Setup -> Utilities -> COP Utilities -> COP Order Integrity Check, we will recalculate the un-picked sales orders total amount. You can, for example, run COP Order Integrity Check over night through defer processing so you know orders that was taken on or before yesterday will have the correct value of order total sales amount when you choose to retrieve that amount through third party application without calculating from the line items yourself.

## ***Mass Delete Attribute***

A new feature is added in Util-Setup -> System File -> Maintenance -> Import -> Mass Delete Attribute to allow users to mass delete attribute records through a CSV file contains the following three columns:

- (A) File Name
- (B) Attribute Code
- (C) Reference ID

## ***General Ledger Improvement***

After you perform a temporary year end in General Ledger, your Balance Sheet will be out of balance until final year end. To solve this issue, we are introducing a new financial code type LYNI (Last Year Net Income) to be used after temporary year end, but before final year end.

## ***Accounts Payable Improvement***

It is a common practice for users to use a manual check to record payments for bank wiring or online ACH payments. Since a unique manual check number is still needed in the AP module to process correctly, a common mistake can take place when a user assigns the manual check number randomly and ends up with a duplication check number. With Elliott V8.5, we now offer an optional AP manual check counter so users can use the F1 key to automatically assign the next manual check number and avoid duplication problems.

In addition, we now offer the option to export A/P check history to a CSV file. Because of the new manual AP check counter feature, we can separate the manual check number range from the computer check. So when we are exporting the A/P check history to CSV, we can easily export only the A/P manual checks records. You can then, for example, interface the CSV file with third party treasury software to process ACH payments for these manual checks.

### ***Exclude Obsolete Customers from F7/F8 Search***

An A/R Global Control setup option was added for “Obsolete Customer Type” to exclude customers with an obsolete customer type from the F7/F8 generic search. User can still use toggle key in the customer generic search to search the obsolete customer if so desired.

### ***Payment History GUI Interface***

We added a Payment History GUI interface that user can access from various areas that can also access open orders or invoice history GUI inquiry. For example, in customer file maintenance or inquiry, you can click on “Special Function” tool bar and choose “Payment History Inquiry”. The payment history inquiry user interface is similar to Open Order/Invoice History Inquiry. The top grid view shows a list of payments (e.g. checks), the bottom grid view shows a list of invoices that payment apply to. Since the current Elliott’s AR Account Inquiry does not offer a way to inquire from payment point of view, it makes this new function very useful.

### ***Improved Cash Receipt User Interface***

The cash receipt user interface had been improved so when users press F3 to display the open invoices that can be applied to, the new windows will display more information, and allow users to pick the invoices to apply easily.

### ***A/R Re-Apply Improvements***

You now have the option to turn on the A/R Re-apply CSV log to keep track of the A/R Re-apply activities. In addition, previously if you made a mistake re-applying an open payment to an invoice, you can’t re-apply it to make that payment open again. We have now improved the application to allow that to happen.

### ***Mass Update Customer by CSV***

In Elliott V8.2, you had the option to make a mass update of the customers’ salesmen through a CSV file. In Elliott V8.5, you now have the option to mass update many other fields in the customer file through a CSV file. This includes customer type, ship via, credit limit, collector, terms code, location, form number, territory, credit rating, freight pay code, comment 1 & 2, user defined notes 1-5, date and amount, account date, statement frequency, transit days, UPS zone, A/R account number, discount %, finance charge, tax code 1-3 and taxable flag.

### ***New Inventory Report Writer Features***

Inventory Report Writer is a special function to export Elliott inventory-related data in item master, inventory location, location 12 months history and ATP files. We created this special function because certain data like -- location 12 months history and ATP -- can be very difficult to retrieve and present in a generic report writer like Crystal or Elliott’s Export Processor.

In Elliott V8.5 we made further improvements to Inventory Report Writer in the following areas: (1) Exporting of all attributes for the item is supported now; (2) Exporting of link document dates is supported now. This is a good way to ensure that links are setup properly for an item master record; (3) There is a separate menu item now for running an Inventory Report only. Also, the inventory report template can be tied to a specific user. This enhances the security so not everyone who can run the inventory report can also modify the report, or run reports designated for other users; (4) Users can now select the items based on the item list in a CSV file; (5) Several fields are added to support EDI 846 (Inventory Advise).

### ***Exclude Zero Quantity Items in F7/F8 Search***

There's a F7 key can be used to exclude zero quantity items in item generic search. It is a toggle key that you can either include or exclude the zero quantity items.

### ***Item Case Size***

In the past, you can use either Item User Amount or Item Order Multiple to stand for item case size. We now added a separate field for Item Case Size. To use it, you will need to go to I/M Global Control setup to indicate which field to be used for box quantity.

### ***Change Customer in Order Entry or Release Held Order***

Since Elliott V8.2, users have the option to change customers during order entry through the "Special Function" toolbar button. This is assuming the user has access to customer file maintenance. However, in many organizations, the salesman does not have the privilege to change customer records because there is sensitive information -- like terms code, credit limit, etc. --that should be not be changed by the salesman.

On the other hand, salesmen should be able change less critical fields like the phone number, contact information, or maybe even address. There is a feature like that in Sales Desk where the admin can define the fields in the customer file that are allowed to be changed through Sales Desk. The same feature is now also available in the Order Entry and the Release Held Order screen, so the salesman does not need to have full customer maintenance access rights in order to change a customer record. Sometimes the credit department also likes the convenience of changing customers' information in Release Held Order.

### ***Auto Release Held Order***

If you enable the automatic credit hold and release feature, then it is likely you will have your credit department review the held orders and decide if the credit situation has been changed and if it is time to release the order. Customers' orders can be placed on hold due to credit reasons like exceeding credit limit, past due, etc. This type of credit issue is usually resolved after the customer sends the payment.

The new Auto Release Held Order feature can be run in defer processing so when the payment is received from customer, this utility can automatically release those orders on credit hold due to credit reasons. This eliminates the need for the credit department to review these orders manually.

## ***Order Edit List Total***

The order edit list is now changed to provide both total based on quantity order and quantity to ship.

## ***Order Inquiry Improvements***

In COP Order Inquiry, user can now use the F4=ChgHst option to see the line item audit trail change history.

Besides the “Date Pick” field, we now provide “\*” or “\*\*” symbols to indicate if the order has been processed for shipping. This includes: (1) Transfer ticket is printed (multi-bin); (2) Transfer batch is completed (multi-bin); (3) Shipping verification is done; (4) Shipping manifest is done for the order.

In addition, in the F3=Ship Data screen, the Box ID (9 digits) is now displayed on the top right corner as each box is highlighted. You can reprint the UCC-128 labels from the Order Inquiry screen.

## ***Mass Billing by CSV Improvements***

In the past, the Mass Billing Selection by CSV Import did not support serialized items. It has been changed to support items with serial numbers, and you can provide the serial number in the CSV file to select the order.

In addition, if you are using Shipment Verification function, you can now optionally import the box information.

## ***Sales Order Import Improvement***

Elliott Sales Order Import is typically use for EDI purposes. It is improved now in the following areas: (1) Support Import Sales Order Line Item Event. For example, you may use this event to subscribe the condition when Ship Via is equal to FedEx and the weight is over 150; (2) Optionally prints warning message when the total sales amount is greater than a certain amount. For example, you may agree with your trading partner that if the order exceeds a certain amount, then you exempt the freight; (3) Optionally prints warning message when the total order weight is greater than a certain amount. For example, when the total order weight is greater than 150, you should change the order to use trucking (LTL); (4) You can now optionally move the import ASCII file with an error to a pre-defined error folder. This will allow you to use deferred processing to run every few minutes to perform the sales order import. If an error takes place, the imported ASCII file is moved to the error folder without being imported again and again with the error in the recurring deferred processing. Optionally, an email can be sent to the admin, warning when an import with an error takes place. (5) You can also optionally move the imported ASCII file without error to a pre-defined folder for archive/auditing purpose. Similar ability is also applied to warehouse receiving import, Mass Billing Import and Inventory Transaction Import. (6) Allow creating sales order export file upon sales order import to send quick turn around EDI 855 Order Acknowledgement. (7) Optionally compares order quantity against ATPQty(99) and give warning. For example, you may set 99 to 3 in global setup. Then during sales order import,

if you do not have enough quantity at the end of next 3 days, then system can give you a warning. Before this, system can only give you warning base on current quantity availability.

### ***Invoice Printing Improvements***

In the past, when print duplicate invoice before the invoice is posted, a new invoice number will be re-assigned. It is now changed that the duplicate invoice will use its original printed invoice number.

Also, system now stores every printed invoice in the CPINVXRF (COP Invoice Cross Reference) table. This includes invoices that are not posted, or even voided invoices. This table also helps us preventing duplicate invoice number. In addition, the system offers an Invoice Log Report under COP -> Processing -> Print Invoices based on the CPINVXRF table.

### ***Require PO Number by EDI Partner***

Generally speaking, most of the EDI trading partners will require PO numbers for you to invoice them. Sometimes your trading partner will place a phone order with you without providing the PO number. Typically, you will have problems invoicing these types of orders through EDI, and that will create various issues for correcting the condition. The best solution will be to capture this condition upon entering the order. You now can specify in the customer's EDI profile that this trading partner requires a PO number. Then, during order entry for this customer, if the PO number is missing, the system will remind the user to request the PO number in order to complete the order.

### ***Shipping Verification Improvements***

Shipping Verification had been improved in the following ways: (1) Some trading partners may require you to place a packing list inside every box you ship to them. Elliott shipping verification can collect box-level information. If you are using shipping verification, and if you choose to print packing lists upon completion of shipping verification, you now have the option to automatically print the number of packing lists based on the number of boxes per shipping verification. (2) You can now optionally create shipping verification box information with Mass Billing by CSV File Import. This may be needed if you use third party software (i.e. WMS) to perform shipping. At a minimum, the box data can be helpful for auditing through with Order Inquiry F3=Ship Data function. In addition, box data can be used to print UCC-128 labels and sending EDI ANS (856). (3) Allow optionally to print UCC-128 labels by item sequence if using VICS BOL function.

### ***Shipment Email Acknowledgement Improvement***

We now support sending shipment email acknowledgement to the Bill-To customer. You can also optionally decide whether to send the tracking number on the email to the Bill-To customer.

### ***Negative Discount Percent***

If you need to set pricing for a customer based on a markup percent, you can now enter a negative discount percent for the customer. This method is different from the traditional markup as supported in Elliott price code, which is based on cost. The negative discount percent method of markup is based on price.

## ***Mass Receiving in Warehouse Receiving***

In the past, you could perform mass PO receiving through the legacy PO receiving process. Mass PO receiving means you can enter the PO number to cause all outstanding PO line items of that PO to be written to the receiving transaction file.

This feature is now available for warehouse receiving as well. However, this feature does not support serialized items. For mass receiving with serialized items, please refer to the improvement below.

## ***Warehouse Receiving CSV Import to Support Serial Number***

In the past, warehouse receiving CSV import did not support serialized items. You now can provide the serial number to the CSV file in warehouse receiving CSV import. Also, you can now optionally move the successfully imported file to a designated folder for archive/auditing purpose.

## ***Support Negative PO Line Item***

The system now supports PO Line Items with negative unit price. You can, for example, use this feature to keep track of deposits on the PO and reduce AP liability when paying the PO.

## ***New Features in Work Order Processing***

Elliott V8.5 Work Order Processing now supports the following new features: (1) Material Shortage Tracking; (2) Production Time Not Captured Report; (3) Allow User to Choose Time Format on Plus Work Order Traveler; (4) Plus Work Order Copy Product Structure; (5) Allow End Item Where-Used Report to Select Items Based on CSV File. (6) If the parent's serial number is same as the component, the system now allows the parent item's serial number to default to component's serial number in material work order production reporting.