



SedonaOffice Release Notes

July 2024

Version 6.2.0.17 (Rev 6) **INTERNAL COPY**

SedonaOffice[®]

Contents

Enhancements	3
34956 Removed Splash Screen from SedonaOffice	3
42126 Miscellaneous Invoices Now Go to the Email Queue [00089931]	3
48745 Added Paging to Inventory Parts Search Page	3
49158 Update to Part_DEL Stored Procedure	4
51363 Changes to EFT Processing.....	4
Application Corrections	5
Accounts Payable	5
30564 Display issue on voided check: amount due and paid incorrect [56521]	5
40272 Attention and Reference lines not populating on Pos [00091642, 96055]	5
42449 Unable to add or select the job number after using the Vendor Lookup/Search button [00095570]	5
46619 Vendor bills for labor going to WIP GL for closed jobs [00093914]	5
51871 Accounts Payable - Payments: Vendor credit not being recognized [118607].....	5
52961 Accounts Payable - Error code: 6 Overflow [00121413]	5
Accounts Receivable.....	6
18808 Autopay Issue: processing date extended 1 month since the update [00009786, 96076, 66228, 64322].....	6
30337 RMR Rate not updating properly and not matching the monthly amount [57771]	6
44554 Transactions marked Previously Funded in Forte show in SedonaOffice as Settled [00099838, 100954, 101104]	6
46002 Batches stuck in the “pending” status [00103404, 103495].....	8
Job Management.....	11
33366 Negative RMR lines do not decrease the RMR [71838]	11
Service	11
44957 Searching for job in job queue and hitting enter does nothing [00062853].....	11
35443 POs on Service Tickets quantity changing to incorrect quantity [00071772]	11
New Stored Procedures	12
Updated Stored Procedures.....	12



New Database Tables12

Special Upgrade/Installation Instructions.....12

Supported Environments13

 Minimum System Requirements.....13

Internal Corrections (not on customer release notes)13

 44554 Transactions marked Previously Funded in Forte show in SO as Settled [Extra internal information]13

 46002 Batches stuck in the “pending” status [00103404, 103495].....17

 46947 Update CentralStations to target Newtonsoft v1319

Enhancements

34956 Removed Splash Screen from SedonaOffice

When users log into SedonaOffice, they would initially see a screen called a splash screen. This splash screen is no longer needed, so we removed it. Now, when users log into SedonaOffice, they go directly to the application.

42126 Miscellaneous Invoices Now Go to the Email Queue [00089931]

Late fee invoices now follow the print or email settings for Miscellaneous Invoices on the Customer Billing screen.

48745 Added Paging to Inventory Parts Search Page

The Inventory Parts screen (Inventory > Parts) could load slowly when the **Pre-Load Parts** checkbox was selected and there were many parts in inventory. To improve the load time, we added paging to the screen at the bottom.

Product Line	Part	Description	Detail	Manufacturer	Primary Vendor	Cost	Sales Price	Costing Meth
Motion Detectors	800PIR	WIRELESS COMMERCIAL PIR	WIRELESS RESID. MOTION	Honeywell	ADI	79.5200	149.54	Standard
Motion Detectors	800PIR-COM	WIRELESS COMMERCIAL PIR S	WIRELESS COMMERCIAL PIR S	Honeywell	ADI	79.5200	194.34	Standard
Motion Detectors	800PIR-ROD	WIRELESS OUTDOOR MOTION	WIRELESS OUTDOOR MOTION	Honeywell	ADI	193.9900	480.13	Standard
Motion Detectors	800PIR-RES	WIRELESS RESIDENTIAL PIR	WIRELESS RESIDENTIAL PIR	Honeywell	ADI	62.8000	153.70	Standard
Motion Detectors	809PPI	WIRELESS FRESH AIR PIR	WIRELESS FRESH AIR PIR	Honeywell	ADI	79.5300	196.64	Standard
Motion Detectors	615SCTN	615SCTN MOTION DETECTOR COLD T	615SCTN MOTION DETECTOR COLD T	Serrotec	ATL	52.1200	129.00	Standard
Motion Detectors	995	RECESS MOUNT PIR MOTION DETECTOR	RECESS MOUNT PIR MOTION DETECTOR	Honeywell	Tri-Ed	38.7900	106.68	Standard
Motion Detectors	995R	RECESS MOUNT PIR MOTION DETECTOR...	RECESS MOUNT PIR MOTION DETECTOR...	Honeywell	Tri-Ed	2.7700	10.67	Standard
Motion Detectors	ALEH450DNF	60 FT INDOOR FLUSHMOUNT BEAM	60 FT INDOOR FLUSHMOUNT BEAM	Alphex	ADI	64.6800	160.09	Standard
Motion Detectors	AP63	LONG RANGE HIRBOR OPTIC PIR MOTION	LONG RANGE HIRBOR OPTIC PIR MOTION	Serrotec	ATL	194.3100	258.17	Standard
Motion Detectors	AP633A	AP633A MOTION DETECTOR LING RING	AP633A MOTION DETECTOR LING RING	Ademco	Tri-Ed	110.8800	274.43	Standard
Motion Detectors	AP730	CURTIN PIR	CURTIN PIR	Ademco	Bosch Security Systems Inc	35.1800	100.00	Standard
Motion Detectors	AJURORA	PET IMMUNE PIR	PET IMMUNE PIR	Honeywell	Communication Connection	16.9600	46.70	Standard
Motion Detectors	AX100S	SURFACE MOUNT MOTION DETECTOR	SURFACE MOUNT MOTION DETECTOR	Optex	Tri-Ed	60.2700	149.17	Standard
Motion Detectors	BO8BLUENEGEN2	TRITECH PIR PIR	TRITECH PIR PIR	Bosch	ADI	106.0000	159.00	Standard
Motion Detectors	BO8DS130	DS150 EXIT MOTION	DS150 EXIT MOTION	Bosch	ADI	61.5000	152.22	Standard
Motion Detectors	BO8DS150T	DS150 EXIT MOTION	DS150 EXIT MOTION	Bosch	Bosch Security Systems Inc	63.2100	156.45	Standard
Motion Detectors	BO8DS160	DS160 EXIT MOTION	DS160 EXIT MOTION	Bosch	Bosch Security Systems Inc	54.2700	134.32	Standard
Motion Detectors	BO8DS161	DS161 EXIT MOTION	DS161 EXIT MOTION	Ademco	Communication Connection	70.6600	174.89	Standard
Motion Detectors	BO8DS415	DS415 PHOTO ELECT BEAM 500 FT	DS415 PHOTO ELECT BEAM 500 FT	Bosch	Bosch Security Systems Inc	88.2600	218.45	Standard
Motion Detectors	BO8DS433	DS433 PHOTO ELEC DETECTOR	DS433 PHOTO ELEC DETECTOR	Bosch	Communication Connection	67.9000	167.07	Standard
Motion Detectors	BO8DS433E	DS433 PHOTO ELEC DETECTOR	DS433 PHOTO ELEC DETECTOR	Bosch	Communication Connection	72.9500	180.56	Standard
Motion Detectors	BO8DS702T	DS05702T	DS05702T	Bosch	ADI	136.0000	338.66	Serial Number
Motion Detectors	BO8DS794Z	LONG RANGE MOTION	LONG RANGE MOTION	Bosch	Communication Connection	93.2300	230.75	Standard
Motion Detectors	BO8DS830	PET MOTION	PET MOTION	Bosch	Communication Connection	30.5800	84.10	Standard
Motion Detectors	BO8DS835E	D5835E PET DUAL MOTION	D5835E PET DUAL MOTION	Bosch	Bosch Security Systems Inc	28.4800	78.32	Standard
Motion Detectors	BO8DS938Z	PANORAMIC PIR W/POPPIT	PANORAMIC PIR W/POPPIT	Bosch	Bosch Security Systems Inc	70.4300	174.32	Standard
Motion Detectors	BO8DS944Z	DS944Z MOTION	DS944Z MOTION	Bosch	Communication Connection	70.0000	172.25	Standard
Motion Detectors	BV300	BV300 MOTION DETECTOR	BV300 MOTION DETECTOR	DSC	ATL	9.3200	30.76	Standard
Motion Detectors	BV300DP	DIGITAL PIR MOTION W/PET IMMUNE	DIGITAL PIR MOTION W/PET IMMUNE	DSC	ATL	11.0600	30.42	Standard
Motion Detectors	BW600	BW600 PET IMMUNE MOTION DETECT	BW600 PET IMMUNE MOTION DETECT	DSC	ATL	23.0200	63.31	Standard
Motion Detectors	BVL4	PET ALLEY LENS FOR BV300 MOTION	PET ALLEY LENS FOR BV300 MOTION	DSC	ATL	2.8200	10.86	Standard
Motion Detectors	CDT7435	DUAL TECH MOTION PIR	DUAL TECH MOTION PIR	Intellisense	Tri-Ed	31.9000	87.73	Standard
Motion Detectors	CDT7435W	DUAL PIR MOTION DETECTOR	DUAL PIR MOTION DETECTOR	Ademco	Communication Connection	75.0000	185.63	Standard
Motion Detectors	CRODD	OUTDOOR DUAL TECH	OUTDOOR DUAL TECH	Crow	Bosch Security Systems Inc	66.4500	164.47	Standard
Motion Detectors	CROWH10	CROW COMH10 PIR	CROW COMH10 PIR	Crow	Communication Connection	16.0300	44.09	Standard
Motion Detectors	CVA2000	CHAMBERLIN WIRELESS MOTION ALERT	CHAMBERLIN WIRELESS MOTION ALERT	Seritex	Tri-Ed	53.2800	131.87	Standard
Motion Detectors	CVPR	WIRELESS MOTION ALERT ADD ON	WIRELESS MOTION ALERT ADD ON	Seritex	Tri-Ed	31.0800	85.47	Standard
Motion Detectors	CW70	LONG RANGE WALL MOUNT PIR	LONG RANGE WALL MOUNT PIR	Optex	ATL	67.0000	165.83	Standard
Motion Detectors	DS415E	DS415E SURFACE PHOTO ELEC BEAM	DS415E SURFACE PHOTO ELEC BEAM	Bosch	ATL	88.0000	217.80	Standard
Motion Detectors	DS936	DS936 MOTION DETECTOR CEILING M	DS936 MOTION DETECTOR CEILING M	Ademco	Tri-Ed	43.5500	119.77	Standard
Motion Detectors	DS9360	PANORAMIC CEILING MOUNT MOTION DET...	PANORAMIC CEILING MOUNT MOTION DET...	Bosch	Tri-Ed	80.1800	198.45	Standard
Motion Detectors	DS938Z	DS938Z MOTION DETECTOR LG CEIL	DS938Z MOTION DETECTOR LG CEIL	Ademco	ATL	68.7800	170.24	Standard
Motion Detectors	DT-500	DT-500 MOTION DETECTOR 35X40 P	DT-500 MOTION DETECTOR 35X40 P	Ademco	Tri-Ed	31.2500	85.94	Standard
Motion Detectors	DT61005TS	MOTION DETECTOR 100/20	MOTION DETECTOR 100/20	Honeywell	Communication Connection	131.6200	325.76	Standard

49158 Update to Part_DEL Stored Procedure

We enhanced the stored procedure called Part_DEL used for deleting parts:

- This no longer allows Part Id 1 to be deleted.
- This no longer allows a part being used in another part's kit to be deleted.
- This now deletes the part's information from the following additional tables. (The deleted part's information previously would be left orphaned in these tables.)
 - IN_Part_GTIN
 - IN_Part_Kit
 - IN_Part_Labor_Unit
 - IN_Part_Price_Level
 - IN_Part_Unit
 - IN_Part_Userdef

We also made corrections to the stored procedure called Part_Supplier_UPD to fix two problems when updating the current record to be the primary vendor:

- This was not properly removing the “primary vendor” indicator from the current primary.
- When copying the primary vendor information from IN_Part_Supplier to the IN_Part record, the vendor part was not getting truncated when it was greater than 25 characters. This caused an error. (The vendor part column in IN_Part is 25 characters, while the vendor part in IN_Part_Supplier is 50 characters.)

We also made corrections to the stored procedure called Part_Supplier_DEL:

- There must be at least one vendor record on a part, so deleting the last vendor record of a part is no longer allowed.
- One vendor record must always be the primary, so now the first (lowest Part_Supplier_Id) vendor record of the part is made the primary if the part's current primary vendor is deleted.

51363 Changes to EFT Processing

For EFT (electronic funds transfer) processing, we added logging to the SEFT_Service_Log:

- When the EFT process determines an orphaned transaction has not been processed by Forte, information is included in the log.
- Results from API search calls are now included in the log.
- Detailed information about failed API search calls is included in the log.

Application Corrections

Accounts Payable

30564 Display issue on voided check: amount due and paid incorrect [56521]

When voiding a check, the system was not displaying the balances and applied amount as expected. We fixed this to display the correct amount due and paid.

40272 Attention and Reference lines not populating on Pos [00091642, 96055]

We fixed the issue of populating both the Attention and Reference address lines on purchase orders when editing the address.

42449 Unable to add or select the job number after using the Vendor Lookup/Search button [00095570]

We enabled the Job Number field and the Job Lookup button when choosing a vendor through the Vendor Lookup/Search button.

46619 Vendor bills for labor going to WIP GL for closed jobs [00093914]

Vendor bills posted for labor on closed jobs were going to the WIP account. We corrected this issue so that vendor bills posted for labor on closed jobs go to the expense account.

51871 Accounts Payable - Payments: Vendor credit not being recognized [118607]

We fixed an error “The amount entered is NOT in balance. Please recheck the items entered.” found when paying a bill and applying a vendor credit; clearing the Print Queue checkbox; and clicking Save.

On the write checks screen:

Bill Amt = Invoice Amount

AmtDue = Amount – Any Credit Applied [Note: previously AmtDue = Amount]

Paid = Amount paid on the invoice [Note: previously Paid = Amount – Amt Due]

Note: This correction requires a database update.

52961 Accounts Payable - Error code: 6 Overflow [00121413]

We increased the size allowed for Invoice IDs on the Write Check/Payment form.

Accounts Receivable

18808 Autopay Issue: processing date extended 1 month since the update [00009786, 96076, 66228, 64322]

The autopay processing date was a month later than it should have been, which was causing accounts to show as late even though the accounts were automatically paid. We fixed this so that cycle invoices no longer adjust the hold date based on the service date.

30337 RMR Rate not updating properly and not matching the monthly amount [57771]

When using the calculator button when creating RMR, users could save recurring lines with monthly and cycle amounts that did not add up correctly. We fixed this issue by properly recalculating the rate based on the RMR amount.

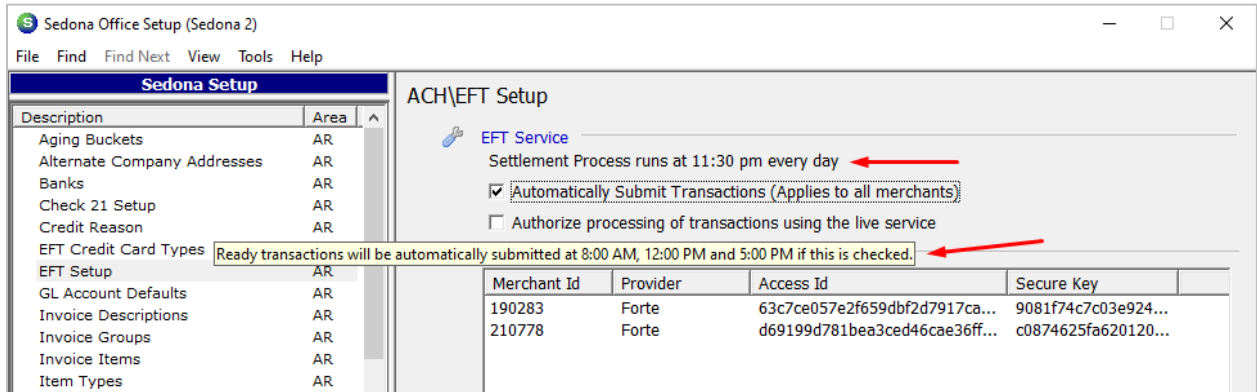
44554 Transactions marked Previously Funded in Forte show in SedonaOffice as Settled [00099838, 100954, 101104]

We enhanced EFT processing and corrected the situation where negative checks were not generating for previously funded transactions:

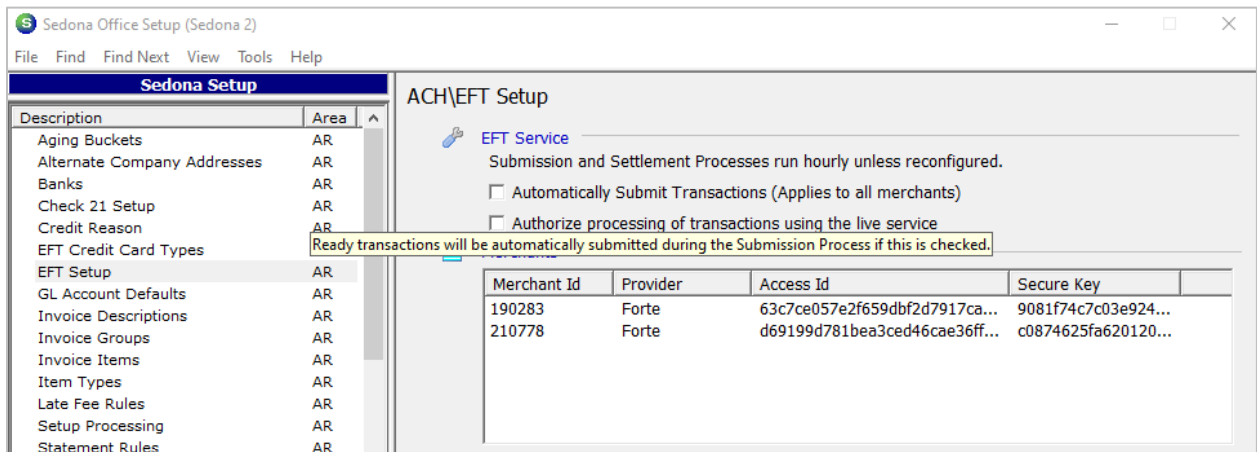
1. We changed the date/time format string used when building the URL for retrieving settlements and fundings from Forte from using a 12-hour clock to using a 24-hour clock, because using a 12-hour clock was causing problems with the 2-hour ranges retrieved, and SedonaOffice was missing some Forte updates.
2. We added the merchant and organization Ids to the SEFT_Reconciliation table. The reconciliation process now tracks reconciliation times by merchant/organization. Previously, if there was more than one merchant/organization in use, they would all update the same reconciliation record. This caused the reconciliation to not have the true times and status for each merchant/organization.
 - We changed the reconciliation process to always include and retrieve the entire date range regardless of how old it is.
 - If there is no reconciliation record for the merchant or organization (essentially meaning the first time it is run), it retrieves the last 90 days of settlements.
3. We changed the fixed processing time to run every hour.
4. The function to delete SEFT_Service_Log lines older than 180 days is done as part of the settlement processing. As settlements can now be run multiple times per day, we changed the delete process so it is run only on the first settlement run of each day.
5. Auto retry upon API call timeout: If a GET API call to Forte times out, and a page size is included in the call (as it is for settlements and fundings), it keeps retrying the call with half the page size until

it succeeds or the page size = 50. All applicable calls currently begin with a page size of 1000. If the call ultimately fails, there is an exception error that is similar if a call initially fails.

6. We changed wording on the EFT Setup form.
 - This was:



- This now says:



7. We added deleting the SEFT_Reconciliation table entries more than 1 year old to the 'once-a-day' processing where the SEFT_Service_Log entries are deleted. The most recent entry for a merchant/organization combination will never be deleted, even if it is more than 1 year old.
8. We moved the check for "Forte update newer than the last update in AR_ACH" to before all status update processing. If the Forte update is not newer than the 'second response date' in the AR_ACH, it skips that Forte update before it does any 'status update' processing.
9. When retrieving settlements, we changed the retrieval to order by the settle_date (instead of by the settle_id). Prior to this, it was possible that status updates would be applied out of order,

causing the SedonaOffice ACH transaction status to end up in an unexpected state and be out of sync with Forte.

10. When retrieving disputes, we changed the retrieval to order by the received_date (instead of by the settle_id). Prior to this, it was possible that status updates would be applied out of order, causing the SedonaOffice ACH transaction status to end up in an unexpected state and be out of sync with Forte.

46002 Batches stuck in the “pending” status [00103404, 103495]

1. After implementing the hourly retrieval of settlements, it was found that our hourly calls to retrieve settlements never received any results. Our transactions would never get updated to ‘settled’, so the batches would remain ‘pending’. After discussing the issue with Forte, they informed us that their transaction timestamps are in Pacific time. Because we used the server’s local time when building the date/time range in the Forte URLs, any server ahead of Pacific time would be asking for transactions in a time range that was partially or completely in the future.

To fix this time zone problem, the EFT service was modified as follows:

- The transaction timestamp time zone was added to the configuration file. This defaults to Pacific time.
- The EFT service determines the local time zone of the server on which it is running.
- The times in the Forte API call URLs are adjusted from local time to the transaction timestamp time zone.

This is an example of what the SEFT log now looks like:

451956	Starting Transactions Settlement Process for Forte merchant 210778 at 2024-02-22 16:41.
451957	Starting settlement process for merchant 210778 - from 2024-02-02 09:42 to 2024-02-22 16:41.
451958	SettleForteTransactions: Retrieve settled transactions for 2024-02-02 07:42 - 2024-02-22 14:41.
451959	MergePagedGetAsync: GET https://sandbox.forte.net/api/v3/organizations/org_356797/locations/loc_210778/settlements/?filter=start_settle_date+eq+2024-02-02T07:42+and+end_settle_date+eq+2024-02-22T14:41&orderby=settle_date&page_size=1000

- Id 451957 shows that the processing is starting for that merchant. Times are computer local time; in this example it is Central Time.
 - Id 451958 shows the date/time range that will be used to retrieve settled transactions from Forte. Times are in “transaction timestamp time zone” time; in this example is Pacific Time.
 - Id 451959 shows the URL of the Forte API call. Notice the dates/times are in Pacific Time.
2. The page_size parameter in the Forte URLs was not getting applied properly. This did not cause any functional problems but has been corrected. (The minimum page size is 50 records.)
 3. In the EFT service, we added the capability to use a random value for “minutes after the hour”, which is now the default option. This accomplishes staggering the time at which EFT processing is performed across the customer base, so that everyone is not hitting Forte at exactly the same

time. When the EFT service is started, the log shows when MINUTES_AFTER_THE_HOUR was randomly generated. If the word random in parenthesis — (random) — is not there, the fixed value from the config file is being used. The transaction time zone is now also included in the log.

4. We fixed the data migration to check if the merchant/organization already has a record, so the migration can be run multiple times on a database.
5. We changed the order of the EFT Service to process orphaned transactions before retrieving settlements. This provides a timelier update of the status of previously orphaned transactions.
6. In the SEFT_Service_Log, additional Forte transaction information is being included for reference and to assist in troubleshooting.
7. We added the AR_ACH_Update_Log table. This stores each settlement transaction from Forte that we process. This is now used in determining if we have already processed a given transaction rather than comparing only the effective date. Before this change, a Forte transaction's effective date had to be after our transaction's last update date to be processed. We found that some Forte transactions have the exact same time stamp as a previous transaction. We were ignoring those, causing our transaction status to get out of sync with Forte. Now, when processing settlements:
 - a. We check the update log to see if the settlement has already been processed. If so, it is simply skipped, and we move on to the next settlement.
 - b. If a Forte transaction's effective date is prior to our transaction's last update date, it is logged to the SEFT service log and we move on to the next settlement. This is an abnormal situation because we should not be getting "new" unprocessed updates from Forte that are prior to our last update. This would indicate a problem with Forte not having updates available in chronological order.
 - c. At this point the Forte transaction effective date is equal to or greater than our last update date and we have not yet processed it. We process it and write it to the update log.

This is also used to track funding settlements that have been processed.

The Sedona EFT Service "once a day" process deletes entries older than one year.

8. During the EFT settlement process, inappropriate status updates are now logged to the SEFT service log and ignored. "Inappropriate" means any status update not appropriate to the transaction's status we currently have in SedonaOffice. For example, if our transaction status is "approved", we expect a status update to be either "settled" or "rejected". If it is not one of those two, we log it as "inappropriate", ignore it, and move on to the next settlement. If our transactions get out of synchronization with Forte, this could help troubleshoot what happened.
9. During the EFT resettle process (going from "previously funded" to "settled"), a SQL query was being executed which would fail on SQL Server versions prior to 2017. This has been corrected.

10. After implementing hourly retrieval of settlements, we found that we were not receiving settlements from Forte in a timely manner. To account for this, we will always retrieve at least 25 hours of settlements. A 24 hour interval will retrieve 48 hours. All other intervals will retrieve 25 hours every <interval> hour(s).
11. If one of our transactions had a status of approved or refunded but got rejected from Forte, our transaction status would get set to previously funded. The status was not obvious to users as being rejected by Forte. And the status of a rejected transaction cannot be changed, while the previously funded status implies the transaction status can be updated in the future.

When updating the status of transactions to previously funded, the SEFT_ACH_UpdateSettlementResponse procedure is setting the response type to Z. Any transaction with a response type of Z is picked up during reversal batch process. We changed transactions rejected by Forte to have a transaction status of rejected (instead of previously funded), but kept the response type set to Z.

12. We found that some settlements from Forte have the exact same timestamp as other settlements for the same transaction. In these cases, we were not necessarily receiving those settlements in the proper order. For example, there could be deposit and withdrawal settlements with a timestamp of 2024-04-05 00:00. If our transaction was in the 'approved' status, we expected the deposit (we were then 'settled') and then the withdrawal (we were then 'previously funded'). However, the settlements were not always sent to us in the correct order. It was possible that we received the withdrawal first (which we ignored as 'inappropriate') and then received the deposit which 'settles' our transaction. At this point we were out of sync with Forte because we should have been 'previously funded'. We added functionality that will reprocess the inappropriate settlements until they are resolved or more than 14 days old. Using the above example, the withdrawal settlement will be reprocessed, and now that our transaction is 'settled', it will become 'previously funded' and we will be in sync with Forte.
13. We found that sometimes settlements were not received by our "get settlements" API call, but were received by our "get funding settlements" API call. This caused our ACH transaction status to not get updated appropriately. For example, our funding batch could get marked as 'complete' by Forte, but transactions inside the batch could still show 'approved' instead of 'settled'. To accommodate this inconsistency, we added functionality to keep retrieving settlements back to the submission date of approved transactions or 14 days, whichever is shorter.
14. Funding settlements are no longer processed until the settlement itself is processed during settlement processing. This ensures an associated deposit check exists and is available to the funding settlement process. Transactions that are only 'approved' will no longer be moved to their funding batch. The transaction must be changed to 'settled' first by the settlement process, then it will be moved by the funding process.

15. We optimized the funding settlement process to better find the deposit check associated with the settlement transaction. This corrects cases where an incorrect deposit check was moved between batches or a deposit check was moved to an incorrect batch.
16. We enhanced error catching when creating a deposit check and processing funding settlements. This provides a more specific and detailed message when errors occur to assist troubleshooting.
17. EFT processing used give an error if a Z-transaction batch total was equal to or greater than or zero. This was incorrect because previously funded *credits* have a positive amount, so it is perfectly acceptable that the total is greater than or equal to zero. We removed this restriction.
18. Related to point 17, EFT processing used to delete a deposit batch once its total reached zero, assuming no deposit checks remained in the batch. Because a deposit batch can contain both positive and negative checks, the total could reach zero before all checks had been removed from the batch if offsetting checks remained. If this happened, the remaining checks would be pointing to a deposit batch that no longer existed. Now, when the total reaches zero, it is verified that there are no checks remaining in the batch before it is deleted.
19. We corrected orphan processing so that a transaction is resubmitted only when the 'search transactions' API call is successful and the results do not contain the transaction being searched. Previously, the transaction would be resubmitted if the transaction was not in the API call results for ANY reason including an exception in the call itself.

Job Management

33366 Negative RMR lines do not decrease the RMR [71838]

When users added recurring through a job and then added a negative change order to decrease the original RMR amount, if they did not enter a negative number in the RMR Amount field, the system did not decrease the RMR even though it showed as a negative amount. Now users must enter a negative RMR Amount to save a Recurring Reversal Change Order.

Service

44957 Searching for job in job queue and hitting enter does nothing [00062853]

We fixed an issue where users were unable to open a job in the selected grid line by pressing Enter on the keyboard. We updated the Job Queue Open list, so users can press Enter on the selected grid line; when enter is pressed, whatever job is currently highlighted in light grey opens.

35443 POs on Service Tickets quantity changing to incorrect quantity [00071772]

We made changes in the AP purchase order form to update the conversion for the expense quantity calculation.

New Stored Procedures

- Deposit_Check_Register_Balance
- Register_List_Balance_Check

Updated Stored Procedures

- SEFT_ACH_UpdateSettlementResponse
- SEFT_NextReconciliationTimings
- SEFT_Reconciliation
- Part_DEL
- Part_Supplier_UPD
- Part_Supplier_DEL

New Database Tables

- AR_ACH_Update_Log

Special Upgrade/Installation Instructions

If upgrading from a SedonaOffice version prior to 6.0, the following related updates are also required:

Legacy SedonaWeb 1.0 — Be aware that if your company uses Legacy SedonaWeb 1.0 (version 2.7.80 or earlier) with SedonaOffice version 6.2.0.8 or earlier, we recommend that you transition to using SedonaWeb 2.0. (Note: Legacy SedonaWeb 1.0 version 2.7.81 is compatible with SedonaOffice 6.2.0.9 or later.)

SedonaWeb/SedonaAPI 2.0 Setup — If your company uses the SedonaWeb/SedonaAPI 2.0 in any manner (Sales Automation, Time & Attendance, eForms, or the SedonaAPI for integrations such as the Manitou integration), IT will update your SedonaWeb/SedonaAPI version at the same time as your SedonaOffice version. This is to ensure compatibility with the Sales Automation module.

Performing Update — Once you have reviewed all the above information, and followed all preparation steps, contact SedonaOffice support. We will note on your account that you have received the Release Notes and are ready for update. SedonaOffice IT will then contact you to schedule your update.

To Use TLS 1.2 — Consider the following:

- All computers running SedonaOffice client must be on Windows 10 with the October 20, 2020 build, version 17763.1554 or later; the server must be on Windows Server 2019 or later.
- TLS 1.2 must be the only TLS version enabled in the Registry. TLS 1.0 and TLS 1.1 must be disabled. Verify they are disabled and TLS 1.2 is enabled.
- The SQL Server must be set to force encryption.

Supported Environments

Minimum System Requirements

- Server is on Microsoft .NET 4.6.1
- If used, SedonaWeb/SedonaAPI 2.0 version 1.47.0 (or higher)

Internal Corrections (not on customer release notes)

These are not in the customer release notes:

44554 Transactions marked Previously Funded in Forte show in SO as Settled [Extra internal information]

We enhanced EFT processing and corrected the situation where negative checks were not generating for previously funded transactions:

1. We changed the date/time format string used when building the URL for retrieving settlements and fundings from Forte from using a 12-hour clock to using a 24-hour clock, because using a 12-hour clock was causing problems with the 2-hour ranges retrieved, and SedonaOffice was missing some Forte updates.
2. We added the merchant and organization Ids to the SEFT_Reconciliation table. The reconciliation process now tracks reconciliation times by merchant/organization. Previously, if there was more than one merchant/organization in use, they would all update the same reconciliation record. This caused the reconciliation to not have the true times and status for each merchant/organization.
 - We changed the reconciliation process to always include and retrieve the entire date range regardless of how old it is.
 - If there is no reconciliation record for the merchant or organization (essentially meaning the first time it is run), it will retrieve the last 90 days of settlements.

3. We changed the fixed processing times of the SedonaEFTService to run at a specified interval:

- We removed the fixed processing times from the SedonaEFTService.exe.config (the SedonaEFTService.exe.config is in the SedonaOffice folder).
- We added HOURLY_INCREMENT, START_HOUR, and MINUTES_AFTER_THE_HOUR to the SedonaEFTService.exe.config:

These are changes to <appSettings> section of the SedonaEFTService.exe.config using HOURLY_INCREMENT, START_HOUR AND MINUTES_AFTER_THE_HOUR.

```
<!--
```

```
HOURLY_INCREMENT specifies how often the  
submission/reconciliation process is run - every  
<HOURLY_INCREMENT> hour(s).
```

```
The default value is 1.
```

```
This must be a value between 1 and 24. Anything outside  
that range will use the default value.
```

```
This must also be a factor of 24 (1, 2, 3, 4, 6, 8, 12,  
24) so the process is run the same number of times, at the  
same time(s), every day.
```

```
If the specified value is not a factor of 24, the default  
value will be used.
```

```
-->
```

```
<add key="HOURLY_INCREMENT" value="1"/>
```

```
<!--
```

```
START_HOUR specifies the first hour of the day that the  
process will run.
```

```
The default value is 0 (midnight).
```

```
This must be a value between 0 (midnight) and 23 (11:00  
PM). Anything outside that range will use the default  
value.
```

For example, if HOURLY_INCREMENT is 4 and START_HOUR is 2, the process will run in the following hours: 2:00 AM, 6:00 AM, 10:00 AM, 2:00 PM, 6:00 PM, 10:00 PM.

If START_HOUR is 3, the process will run in hours 3:00 AM, 7:00 AM, etc.

If START_HOUR is greater than or equal to HOURLY_INCREMENT, START_HOUR will be reduced by HOURLY_INCREMENT until it is an appropriate value.

For example, a 2-hour increment cannot be started in the 5:00 hour - it must be started between midnight and 1:59 AM (this value would be 0 or 1).

In this scenario, START_HOUR will be reduced by 2 hours until it is less than 2 (5 minus 2 minus 2), resulting in 1.

If HOURLY_INCREMENT is 1, START_HOUR is irrelevant and ignored as the process will run every hour, starting in the midnight hour.

-->

```
<add key="START_HOUR" value="0"/>
```

<!--

MINUTES_AFTER_THE_HOUR specifies the minute at which the process will begin.

The default value is 15.

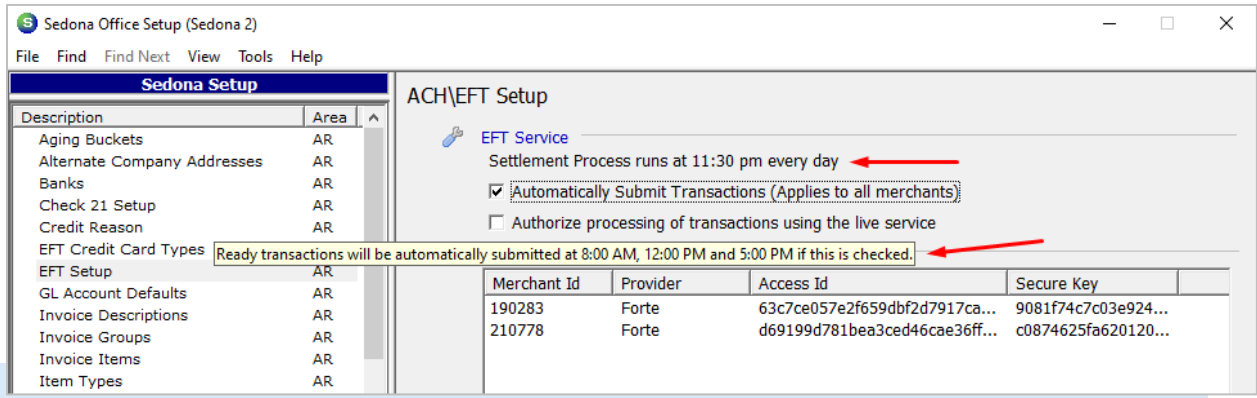
This must be a value between 0 and 59. Anything outside that range will use the default value.

For example, if HOURLY_INCREMENT is 4, START_HOUR is 2, and MINUTES_AFTER_THE_HOUR is 20, the process will run at 2:20 AM, 6:20 AM, 10:20 AM, 2:20 PM, 6:20 PM, 10:20 PM.

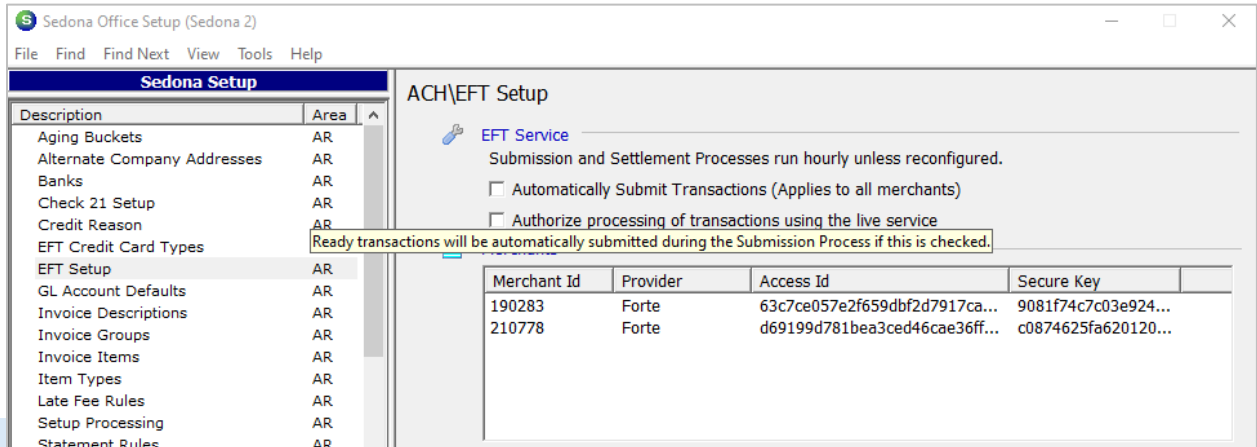
```
-->
<add key="MINUTES_AFTER_THE_HOUR" value="15"/>
```

- Settlements are no longer retrieved in 2-hour chunks but are retrieved with the full applicable date/time range. If the results cannot be obtained before a timeout occurs, the system automatically retries. This should increase the efficiency of the data retrieval, and the SedonaOffice transactions status updates can occur throughout the day rather than once overnight (depending on hourly increment).
 - When the settlement process is run, the beginning time of the range is extended by the hourly increment. For example, given a 2-hour increment, the start time as obtained from SEFT_Reconciliation should be 2 hours earlier (the end time of the previous run). The process goes back another 2 hours (the hourly increment) for 4 hours total.
 - By default, the process runs once an hour, meaning it retrieves only 2 hours.
 - The maximum hourly increment is 24, meaning it would retrieve 48 hours of settlements once a day.
- 4. The function to delete SEFT_Service_Log lines older than 180 days is done as part of the settlement processing. As settlements can now be run multiple times per day, we changed the delete process so it is run only on the first settlement run of each day.
- 5. Auto retry upon API call timeout: If a GET API call to Forte times out, and a page size is included in the call (as it is for settlements and fundings), it keeps retrying the call with half the page size until it succeeds or the page size = 50. All applicable calls currently begin with a page size of 1000. If the call ultimately fails, there is an exception error that is similar if a call initially fails.
- 6. We changed wording on the EFT Setup form.

• This was:



- This now says:



7. We added deleting the SEFT_Reconciliation table entries more than 1 year old to the 'once-a-day' processing where the SEFT_Service_Log entries are deleted. The most recent entry for a merchant/organization combination is never be deleted, even if it is more than 1 year old.
8. We moved the check for "Forte update newer than the last update in AR_ACH" to before all status update processing. If the Forte update is not newer than the 'second response date' in the AR_ACH, it skips that Forte update before it does any 'status update' processing.
9. When retrieving settlements, we changed the retrieval to order by the settle_date (instead of by the settle_id). Prior to this, it was possible that status updates would be applied out of order, causing the SedonaOffice ACH transaction status to end up in an unexpected state and be out of sync with Forte.
10. When retrieving disputes, we changed the retrieval to order by the received_date (instead of by the settle_id). Prior to this, it was possible that status updates would be applied out of order, causing the SedonaOffice ACH transaction status to end up in an unexpected state and be out of sync with Forte.

46002 Batches stuck in the "pending" status [00103404, 103495]

This is INTERNAL information only related to 46002:

This is how the changed configuration items now look:

```
<appSettings>
<!--
HOURLY_INCREMENT specifies how often the submission/reconciliation process is run - every <HOURLY_INCREMENT> hour(s).
The default value is 1.
This must be a value between 1 and 24. Anything outside that range will use the default value.
This must also be a factor of 24 (1, 2, 3, 4, 6, 8, 12, 24) so the process is run the same number of times, at the same time(s), every day.
If the specified value is not a factor of 24, the default value will be used.
-->
<add key="HOURLY_INCREMENT" value="1"/>

<!--
START_HOUR specifies the first hour of the day that the process will run.
The default value is 0 (midnight).
This must be a value between 0 (midnight) and 23 (11:00 PM). Anything outside that range will use the default value.
For example, if HOURLY_INCREMENT is 4 and START_HOUR is 2, the process will run in the following hours: 2:00 AM, 6:00 AM, 10:00 AM, 2:00 PM, 6:00 PM, 10:00 PM.
If START_HOUR is 3, the process will run in hours 3:00 AM, 7:00 AM, etc.
If START_HOUR is greater than or equal to HOURLY_INCREMENT, START_HOUR will be reduced by HOURLY_INCREMENT until it is an appropriate value.
For example, a 2-hour increment cannot be started in the 5:00 hour - it must be started between midnight and 1:59 AM (this value would be 0 or 1).
In this scenario, START_HOUR will be reduced by 2 hours until it is less than 2 (5 minus 2 minus 2), resulting in 1.
If HOURLY_INCREMENT is 1, START_HOUR is irrelevant and ignored as the process will run every hour, starting in the midnight hour.
-->
<add key="START_HOUR" value="0"/>

<!--
MINUTES_AFTER_THE_HOUR specifies the minute at which the process will begin.
The default value is -1.
This must be a value between -1 and 59. Anything outside that range will use the default value.

A value of -1 will cause the startup process to select a random minute from 0 - 59 each time the EFT service is started. Note that this is *not* each time the
submission/reconciliation process is run once the service is started - the selected random minute is used until the service as a whole is stopped.
Using a random minute will stagger the EFT processing across the customer base, so that everyone is not hitting the processor at exactly the same time.

A value of 0 - 59 will use that minute every time the EFT service is started.
For example, if HOURLY_INCREMENT is 4, START_HOUR is 2, and MINUTES_AFTER_THE_HOUR is 20, the process will run at 2:20 AM, 6:20 AM, 10:20 AM, 2:20 PM, 6:20 PM, 10:20 PM.
-->
<add key="MINUTES_AFTER_THE_HOUR" value="-1"/>
</appSettings>
```

```
<!--
TRANSACTION_TIMESTAMP_TIME_ZONE specifies the time zone in which the transaction time (settlement date, funding effective date, etc.) resides.
The default value is Pacific time.
The value must be in the exact format as shown in the time zone dropdown in Date and Time settings in Windows setup, except that an ampersand must be
replaced by &#03B; as this configuration file is XML.
If a valid time zone is not specified, the default value is used.
When retrieving transactions by date/time range, the dates/times in the URL are adjusted from the local computer time to this "transaction timestamp time zone".
Example:
- The computer running the EFT service is in Eastern.
- The transaction timestamp time zone is Pacific.
- An hourly run of the reconciliation process runs at 12:00 noon.
- The reconciliation process wants transactions from 10:00-12:00 (Eastern).
- The URL time range will have 07:00-09:00 (Pacific).
(UTC-08:00) Pacific Time (US &#03B; Canada)
-->
<add key="TRANSACTION_TIMESTAMP_TIME_ZONE" value="(UTC-08:00) Pacific Time (US &#03B; Canada)"/>

<!--
MINIMUM_RETRIEVAL_HOURS specifies the minimum number of hours that will be included in a reconciliation date/time range.
The default value is 2.
This must be a value between 2 and 168. Anything outside that range will use the default value.

Because settlements are not always received in a timely manner, this option provides the means to retrieve settlements with a timestamp further in the past
than the hourly increment normally retrieves. If a reconciliation range includes less hours than this value, the start date/time will be adjusted back so
the range includes this many hours.

For example, an hourly increment will normally retrieve the past 2 hours every hour. If this value is 24, an hourly increment will retrieve the past 24
hours every hour.
-->
<add key="MINIMUM_RETRIEVAL_HOURS" value="25"/>

<!--
MAXIMUM_RETRIEVAL_DAYS specifies the maximum number of days that will be included in a reconciliation date/time range.
The default value is 14.
This must be a value between 7 and 30. Anything outside that range will use the default value.

The reconciliation start date/time is not bumped when a reconciliation run is unsuccessful, which could go on indefinitely if consecutive runs are unsuccessful.
This option prevents a runaway retrieval range where the start date/time stays the same while the end date/time continues to move forward. If the retrieval
range exceeds the maximum retrieval days, the start date/time is increased by the hourly increment.

This option is essentially saying, "If we have not received and/or successfully processed a settlement this old by now, we probably never will. Move on."
-->
<add key="MAXIMUM_RETRIEVAL_DAYS" value="14"/>
</appSettings>
```

Although not documented in the config file, it should be noted that the transaction time zone can also be specified by what Windows calls the "time zone ID" (example = "Pacific Standard Time"). The config file is

specifying what Windows calls the "display name" of the time zone. Also allowing the "time zone ID" is intended to accommodate full compatibility in case a version of Windows or .NET does not return the "display name" of the time zone to our program as "(UTC-08:00) Pacific...". Hopefully if this is ever the case, the "time zone ID" will still be as expected.

To see the time zones available on a computer, the command "tzutil /l" can be entered at a command prompt.

```
(UTC-08:00) Baja California  
Pacific Standard Time (Mexico)  
  
(UTC-08:00) Coordinated Universal Time-08  
UTC-08  
  
(UTC-08:00) Pacific Time (US & Canada)  
Pacific Standard Time  
  
(UTC-07:00) Arizona  
US Mountain Standard Time  
  
(UTC-07:00) La Paz, Mazatlan  
Mountain Standard Time (Mexico)  
  
(UTC-07:00) Mountain Time (US & Canada)  
Mountain Standard Time  
  
(UTC-07:00) Yukon  
Yukon Standard Time  
  
(UTC-06:00) Central America  
Central America Standard Time  
  
(UTC-06:00) Central Time (US & Canada)  
Central Standard Time
```

This shows both the "display name" and the "time zone ID" of each time zone. The configuration value can be set to either of these.

If the specified time zone is not found, the program will attempt to use time zone "Pacific Standard Time". It uses the "time zone ID" in case the "display name" is "(UTC-08:00)..." but was not found. This may never happen, but would allow the program to continue on if it ever did happen. If Pacific time is not the desired time zone, it must be ensured that the exact "display name" or "time zone ID" of the desired time zone is specified in the configuration value.

46947 Update CentralStations to target Newtonsoft v13

Unified solution projects were targeting newtonsoft.json.dll version 13, while the CentralStations solution target was version 12. This caused DLL version mismatch errors when attempting to communicate with Central Stations for integration functionality.

We retargeted Central Station DLLs that targeted Newtonsoft.Json.dll v12 to target v13.

We retargeted Unified projects to latest nugets (1.0.0.9) from VBcompatiblePerennialShared.

We also ensured all other nuget packages common between VBcompatiblePerennialShared and Unified target the same version.

This affected these DLLs:

- AddressVerificationWeb.dll
- CentralStationConnector.dll
- CentralStationController.dll
- CentralStations.dll
- CentralStations.Stages.dll
- HarnessCentralStationConnector.dll