

POSCONNECT DEFERRED PROCESSING DESIGN SPECIFICATION

OVERVIEW

To further improve the performance of POSConnect and its response time, POSConnect offers a new fully user configurable feature called *Deferred Processing (A.K.A DP)*. The Deferred Processing enables POSConnect to distribute the Transaction Life Cycle in to two separate components:

- ➡ Fund Verification & Authorization – processed interactively
- ➡ Posting to F/A and A/R – performed concurrently in the background

When DP is enabled the response time for PURCH will always be the same as the approval time which is the same as performing an INQ (A.K.A “BI”) transaction. Deferred Processing will eliminate the performance degradation issues and bottlenecks caused by client custom subroutines such as FA Transmittal or over populated data such as A/R invoices causing database to run slow.

HIGHLIGHTS

1. **PREPURCH** - The current Fund verification & authorization & posting subroutine – called XS.TCSPOS.PREPURCH, hereafter referred to as PREPURCH. The PREPURCH process checks funds verification and then posts the transaction if the deferred processing option is disabled. When the deferred processing option is enabled, PREPURCH writes a transaction into a queue to be processed by the Deferred Processing Listener after the approval and returns the control back to the Java Listener with a *success* message without advancing to the FA and AR invoices and payments posting phase.
2. **BGPURCH** - The background posting process – called XTCSP0SB01 and XTCSP0SB02 , hereafter referred to as BGPURCH, performs the same purchasing function as XS.TCSPOS.PREPURCH with the addition of an output transaction file that contains the results of each deferred purchase transaction processed and an input queue file that contains the transactions that are queued up and waiting to be processed.
3. **XDSP1** – The DP Command Center. This form’s function is starting and stopping the Deferred Processing Listener.
4. **XTCSP0SR02** - Deferred Processing Log Report – prints transactions processed by the BGPURCH.
5. **XDPBP** - Batch process – to process the transaction queue in the event the DP Listener is failing or to be used generally as an alternate or backup mode to the DP Listener.

SUMMARY

When Deferred Processing is enabled, after the transaction is approved PREPURCH will write the transaction to a queue to immediately be picked up and processed by the Deferred Processing Listener allowing the student to complete his/her transaction before the posting has been processed, hence, quickly exiting the Cashier area. The DP Listener completes the posting phase (BGPURCH) concurrently. The form XDSP1 is used to start or stop the deferred process Listener, a new UniData/UNIX process that performs the processing and logs the transaction results. BGPURCH and PREPURCH both include the same Fund Verification logic.

POSCONNECT DEFERRED PROCESSING DESIGN SPECIFICATION

REQUIREMENTS & ASSUMPTIONS

UniData Development - Minimal coding and minimal work done by UniData resources.

XML Schema - The existing XML Schema and PREPURCH transaction type will remain unchanged in order to not trigger any customization from the POS vendor. The PREPURCH subroutine will return the exact same results regardless of the background processing option setting.

Refunds - Refund transactions, which are also PREPURCH transaction types with a negative dollar balance, and are also subject to Deferred Processing.

POSCONNECT DEFERRED PROCESSING DESIGN SPECIFICATION

PROPOSED SOLUTION – *Java Component*

XML

No XML modifications required.

POSConnect.properties

The following new configuration parameter is added to the POSConnect.properties file:

TCSPoS.deferred.processing=true/false

☐ Each listener has its own properties file.

Com.premieresys.TCS.POS.TCSPOSListenerConfig.java

Modifications to include the new configuration file option as new variables.

Com.premieresys.ServiceProvider.Services.StandardUnidataSubroutineService.java

Modifications to private ServiceOutput invokeSubroutine(String record_IN). It will read the TCSPoS.deferred.processing configuration variable and pass it to the PREPURCH subroutine.

POSConnect Listeners

Pseudo Code

IF PURCH THEN

 Insert TCSPoS.deferred.processing setting value as input argument 18 for the
 XS.TCSPoS.PREPURCH subroutine.

END

Process regular transaction

PROPOSED SOLUTION – *Envision Component*

XS.TCSPoS.PREPURCH

The current XS.TCSPoS.PREPURCH subroutine will be modified to ignore an obsolete queuing function controlled from XPRM screen called “Use XTHT Queue” using attribute 18 of the input argument XIN.REQUEST. The process flow remains the same with the following exception: After calling the XS.TCSPoS.PURCHASE.REFUND subroutine, if attribute 18 is true, “Y”, and the response argument XOUT.RESPONSE attribute 9 is null the entire XIN.REQUEST argument is written to the transaction queue **XTCSPOS.DEF.TRN.FILE** for processing by the deferred processing phantom.

POSCONNECT DEFERRED PROCESSING DESIGN SPECIFICATION

Pseudo Code

The process flow of XS.TCSPOS.PREPURCH remains the same until returning from the call to the subroutine XS.TCSPOS.PURCAHSE.REFUND. At that point, XS.TCSPOS.PREPURCH checks the 18th attribute of the argument XIN.REQUEST and the 9th attribute of XOUT.RESPONSE.

- If XIN.REQUEST<18> is true and XOUT.RESPONSE is null then do the following:
 - Assign XIN.REQUEST” to the variable XL.STUFF.
 - In XL.STUFF replace @FM with “|”.
 - In XL.STUFF replace @VM with “&”.
 - In XL.STUFF replace @SM with “^”.
 - Write the transaction contained in XL.STUFF to the deferred transaction queue XTCSPOS.DEF.TRAN.FILE as follows:
 - Assign invoice number from the transaction contained in the argument XIN.REQUEST to the transaction queue ID, XTCSPOS.DEF.TRAN.FILE.ID.
 - Assign the variable to the data element XDIF.DATA.
 - Write the record using commit and new as the options.
- Continue the normal process flow already established in XS.TCSPOS.PREPURCH.

XS.TCSPOS.PURCHASE.REFUND

The current XS.TCSPOS.PURCHASE.REFUND subroutine will be modified to check for the deferred processing flag attribute 18 of input argument AL.REQUEST. If it's anything except true “Y”, it will operate normally. If it's set to “Y”, the subroutine will process normally but will not create an invoice by calling subroutine S.CREATE.AR.INVOICE nor call the posting subroutine XS.TCSPOS.POST.PURCH. The subroutine creates a new transaction status of “PENDING” in **XTCSPOS.TRANS.LOG** with @ID = STUDENT_ID*FUND_SOURCE_ID*INVOICE_NO*INVOICE_DATE*PURCH_AMT (this is a POSConnect Log file and is not specific to Deferred Processing) and a subsequent new error message of "3004. This transaction is deferred and PENDING" when checking for duplicate transactions.

Pseudo Code

The process flow of XS.TCSPOS.PURCAHSE.REFUND remains the same with the following exceptions:

- Assign attribute 18 of the argument AL.REQUEST to the variable X.DEFERRED.PROCESSING.FLAG.
- After calling internal subroutine X.CHECK.TRANS.STATUS and external subroutine XS.TCSPOS.TRANS.STATUS if the returned status attribute 7 contains “PENDING” and X.DEFERRED.PROCESSING.FLAG is “Y” assign “1” to the variable X.STATUS.ERR and "3004. This transaction is deferred and PENDING" to X.RES.MSG returning to the calling process.

POSCONNECT DEFERRED PROCESSING DESIGN SPECIFICATION

- If X.DEFERRED.PROCESSING.FLAG is "Y" by pass invoice creation, updating SA.MBD.USED and posting of the transaction.
- Continue the normal process flow already established in XS.TCSPOS.PURCAHSE.REFUND.

POSCONNECT DEFERRED PROCESSING DESIGN SPECIFICATION

XTCSPOSB01

Is a new process that mimics the process workflow of XS.TCSPOS.PREPURCH except there is no output argument, input is from a queue file and a transaction log file of records process and the results returned. The transaction log file is keyed by student ID, date and time the contents of the transaction log record response field are:

- Type of transaction
- Student ID
- Fund Source
- Term ID
- Invoice number
- Transaction date
- Transaction amount
- Transaction acknowledgement 1 for success 0 for failure
- Transaction message if any. Blank if the transaction was successfully processed or the error message if the transaction failed.

Pseudo Code

This process is started and stopped by form XSDP1.

The process flow is as follows:

- Start an infinite loop.
 - Select the transaction queue.
 - Read each selected record, assign the results to variable XL.RECORDS and delete the record from the queue.
 - Loop through each record stored in XL.RECORDS processing as follows:
 - Swap "|" with @FM in X.RECORD.
 - Swap "&" with @VM in X.RECORD.
 - Swap "^" with @SM in X.RECORD.
 - Assign X.RECORD to XIN.REQUEST.
 - Execute the code from subroutine XS.TCSPOS.PREPURCH without queuing.
 - Update the transaction log with the results returned from XS.TCSPOS.PREPURCH.
 - After processing all records contained in XL.RECORDS, exit the inner loop.
 - Put the process to sleep for 10 seconds.
- Repeat the outer loop until the phantom is manually stopped by XDSP1.

POSCONNECT DEFERRED PROCESSING DESIGN SPECIFICATION

XTCSPOSB02

Is a new process that mimics the process workflow of XS.TCSPOS.PREPURCH except there is no output argument, input is from a queue file and a transaction log file of records process and the results returned. The transaction log file is keyed by student ID, date and time the contents of the transaction log record response field are:

- Type of transaction
- Student ID
- Fund Source
- Term ID
- Invoice number
- Transaction date
- Transaction amount
- Transaction acknowledgement 1 for success 0 for failure
- Transaction message if any. Blank if the transaction was successfully processed or the error message if the transaction failed.

Pseudo Code

This process is executed by form XDPBP.

The process flow is as follows:

- Start an infinite loop.
 - Select the transaction queue.
 - Read each selected record, assign the results to variable XL.RECORDS and delete the record from the queue.
 - Loop through each record stored in XL.RECORDS processing as follows:
 - Swap "|" with @FM in X.RECORD.
 - Swap "&" with @VM in X.RECORD.
 - Swap "^" with @SM in X.RECORD.
 - Assign X.RECORD to XIN.REQUEST.
 - Execute the code from subroutine XS.TCSPOS.PREPURCH without queuing.
 - Update the transaction log with the results returned from XS.TCSPOS.PREPURCH.
 - After processing all records contained in XL.RECORDS exit the inner loop.
 - Put the process to sleep for 10 seconds.
- Repeat the outer loop until the phantom is manually stopped by XDSP1.

POSCONNECT DEFERRED PROCESSING DESIGN SPECIFICATION

XTCSPOSR02

Is a new process that reports the transaction results from the deferred processing of transactions or from the batch processing of deferred transactions.

Pseudo Code

This process is started by form XDPR.

The report is generated by the Envision reporter as follows:

- Header data consist of:

Fld#	Usg	Lin	Ord	Zone	Text
1	H	1	1	L	Date: \D2/
2	H	1	2	C	POSCONNECT Deferred Processing Log Report
3	H	1	3	R	Page: \P4
4	H	2	1	L	

- Column data consists of:

Fld	Cde	Field Name	Usg	Lin	Col	Tot	Wid	Sep	Jst	Conv
1	a	XDTL.TIME	CD	1	1	1	8	1	R	MTS
2	b	XDTL.ACCT.ID	CD	1	2	2	10	1	L	
3	c	FIRST.NAME	CD	1	3	3	20	1	L	
4	d	LAST.NAME	CD	1	4	4	25	1	L	
5	e	ID					10		R	
6	f	XDTL.RESPONSE	CD	1	5	5	60	1	L	

- Print the detail using this statement PRINT_DETAIL ITEMIZING XDTL.RESPONSE
- The Report format will appear as follows:

```

H 1|Date: 11/29/11                POSCONNECT Deferred Processing Log Report                Page: nnn
H 2|
H 3|
CH 1| Time Student ID First      Last Name      Response
CH 2|-----
CD 1|aaaaaaaa bbbbbbbbbb ccccccccccccccccccc ddddddddddddddddddddd ffffffffffffffffffffffffffffffffffffff
CDS 1|                                ffffffffffffffffffffffffffffffffffffff
CD 2|
    
```

POSCONNECT DEFERRED PROCESSING DESIGN SPECIFICATION

FILES USED IN DEFERRED PROCESSING

XTCSPOS.DEF.TRN.FILE File Layout – AKA “The DP Queue” used by the Deferred Processing Listener

Data Element	Usage	Position	Internal Size	Display Size	Justification	Conversion	Association	Purpose
XTCSPOS.DEF.TRN.FILE.ID	K	1	10	10				Key to the file. Invoice number received from the POS system.
XDTF.DATA	D	1	200	200	L			Transaction received from the POSCONNECT listeners with @FM, @VM, @SM converted to “ ”, “g”, “^” and stored as a single string of data.

XTCSPOS.DEFERRED.LOG File Layout – Log file updated by the Deferred Processing Phantom

Data Element	Usage	Position	Internal Size	Display Size	Justification	Conversion	Association	Purpose
XTCSPOS.DEFERRED.LOG.ID	K	1	12	12	L			Current Date in internal format
XDTL.TIME	K	2	8	8	R	MTS		Current Time in internal format
XDTL.ACCT.ID	K	3	10	10	L			Student ID
XDTL.RESPONSE	L	1	200	200	L			Actual response that would be returned to the POSConnect listeners.

Future Enhancements & Bug Fixes:

- ☞ The DP Listener frequency should be configurable by the user via XPRM screen.
- ☞ XTCSPOSB01, XTCSPOSB02, and XS.TCSP0S.PREPURCH should not be 3 separate physical programs as they all identical.
- ☞ Users should be able to configure multiple DP queues using XPRM screen.
- ☞ DP Listener should not allow any transaction for a given student if the student has any PENDING transactions. This will produce a result as a record lock for Bookstore purchases only.
- ☞ DP should be able to check for duplicate invoices. Refer to Test 7.7.
- ☞ DP should be able to check for valid dates. Refer to Test 7.16.
- ☞ Should not create a record in XTCSPOS.DEF.TRN.FILE for Unapproved transactions. Refer to Test 7.13.