



# IBM Sterling Order Management – Return Enhancements

# Typical Retail Return Scenario



*"It's a disgrace! I've only had this for two weeks, and LOOK! The novelty has almost completely worn off!"*



"He wants to return some shade trees because they stopped working in November."

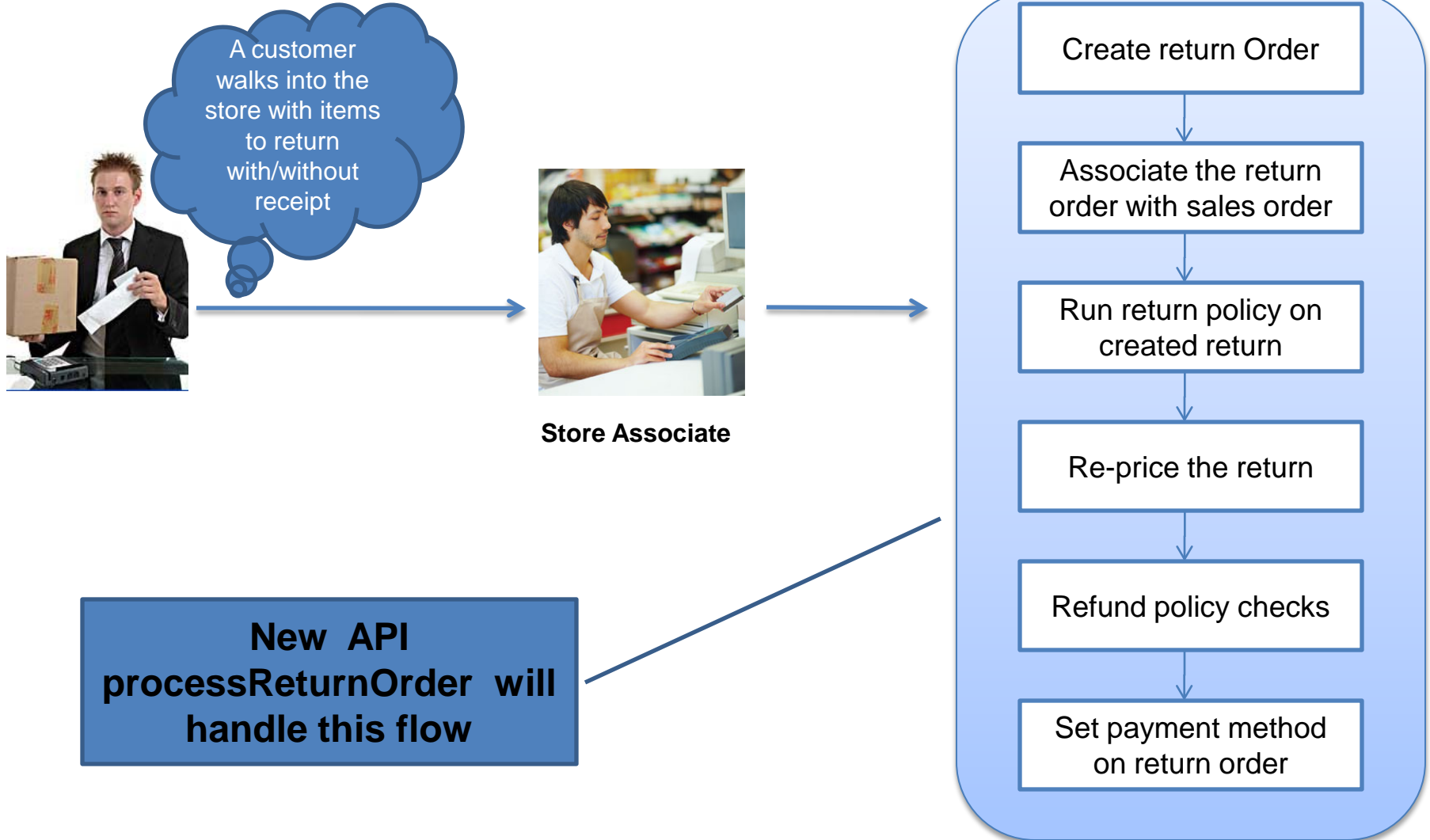
Handling complicated return scenarios, required robust return mechanism in OMS

# Return Module Enhancements

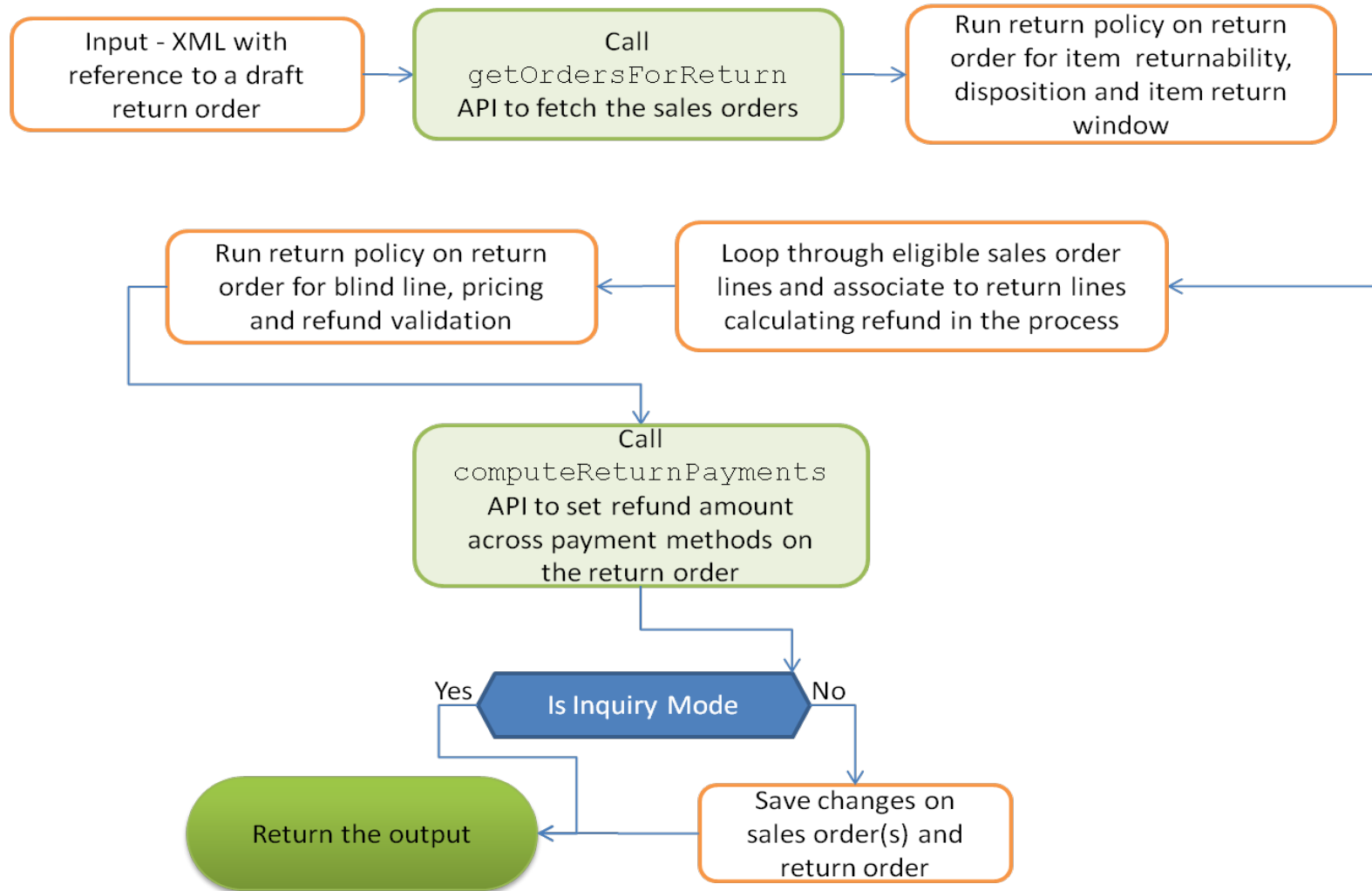


- The return module is enhanced to handle the blind returns scenarios for POS/store
- New API introduced to associate the sales order with return order in case customer doesn't have receipts
- Return policy checks on the return order created
- Re-pricing of the return based on sales order associated
- Refund amount policy checks
- Setting payment method on the return order

# High Level Business Scenario



# processReturnOrder API flow



# Return Order To Sales Order Association (1)



- Return order has to be associated to sales order in case where customer doesn't have receipts
- Sales order associated based on some key attributes of sales order like customer , payment information
- The API processReturnOrder in turn calls getOrdersForReturn to fetch the sales order
- The results of getOrdersForReturn will be further to filtered to associate the return order with sales order

# Return Order To Sales Order Association (1)

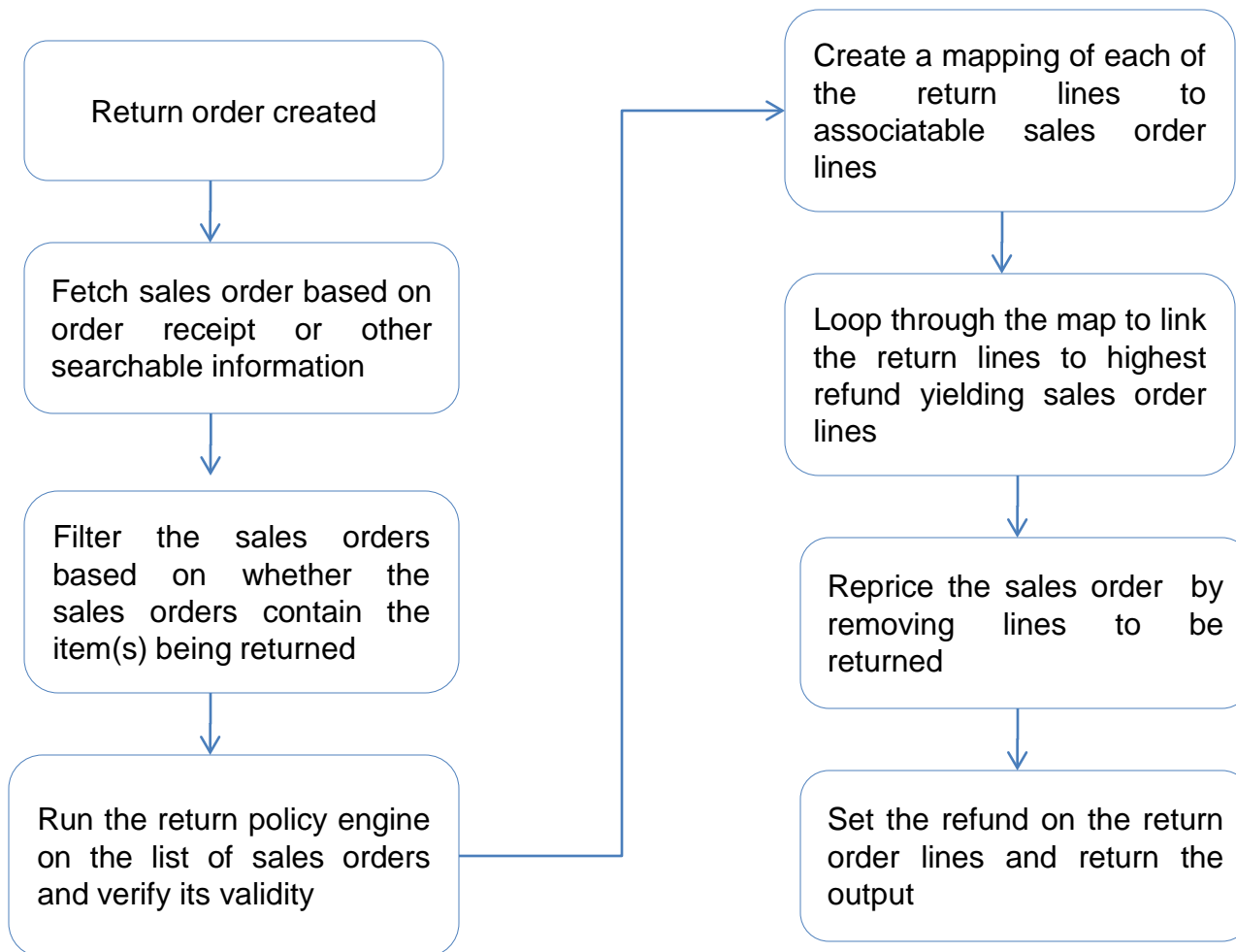
## Order search criteria available

If customer doesn't have receipt, below order search criteria is allowed.

- **Credit Card:** DisplayCreditCardNo, CreditCardExpDate, CreditCardName, CreditCardType, FirstName, MiddleName, LastName, CreditCardNo
- **SVC:** DisplaySvcNo, SvcNo
- **Customer Account:** DisplayCustomerAccountNo, CustomerAccountNo
- **Other:** DisplayPaymentReference1, CheckNo, PaymentReference1, PaymentReference2, PaymentReference3

# Return Order To Sales Order Association (2)

## processReturnOrder API Flow





# Return Order To Sales Order Association (3)

## ILLUSTRATION -1

### Original Sales Order

Item ID	Quantity	Unit Price
Item1	10	20
Item2	5	10
Item3	3	5

### Items to be returned

Item ID	Quantity	Associated To Sales Order
Item1	15	N
Item2	3	N

After running processReturnOrder API



Item ID	Quantity	Associated To Sales Order
Item1	10	Y
Item1	5	N
Item3	3	Y

Blind line validation will be run on the line 2 that was created with Item1 and quantity 5.

# Return Order To Sales Order Association (4)

## ILLUSTRATION -2

### Sales Order- 1

Item ID	Quantity	Unit Price
Item1	10	20
Item2	5	10
Item3	3	5

### Sales Order- 2

Item ID	Quantity	Unit Price
Item1	10	20
Item2	5	10
Item3	3	5

The Order 2 is picked for the return as the refund amount for return order will be maximum

### Items To be returned

Item ID	Quantity	Associated To Sales Order
Item1	15	N
Item2	3	N

### On call to Process return Order

Item ID	Quantity	Associated To Sales Order
Item1	10	Y ( Order1)
Item 1	5	Y (Order 2)
Item 2	3	Y (Order 2)

# Return Policy Rules(1)

- All the sales order returned in getOrderForReturn API will be further filtered based on return policy rules
- New validations are added validation repository for return processing
- The validation are added to **Returns Validation and Return pricing domains**
- Each of these *domains* will have a set of *validation* which will in turn contain detection mechanism
- The detection mechanism detect whether or not override rules need to be run for the validation
- Based on the call whether to run the override rules for the input, checkOverridenRule API will be called
- These validation checks may further lead to violations that need to be approved by a manager

# Return Policy Rules(2)

Below return validations are performed as part of processReturnOrder API

Rule Type	Description
Return Window	check if the sales orders to associate with the return line are within the return window range.
Return Reason	This validation will check the Return Reason Code of the Line being returned
Is Returnable	This validation can be used to accept/reject an item which is marked as not returnable.
Unit Price Limit	This validation will used to check for a unit refund value for an item on an order line.
Line Price Override Limit	The MANUAL_PRICE_OVERRIDE validation for Sales orders will be reused for the return order document type too
Blind Line	This validation is used to check whether the creation of the blind line is allowed by the enterprise.

# Return Policy Rules(3)

## Rules Executed BEFORE Re-pricing return

Rule Type	Description
Return Window	check if the sales orders to associate with the return line are within the return window range.
Return Reason	This validation will check the Return Reason Code of the Line being returned
Is Returnable	This validation can be used to accept/reject an item which is marked as not returnable.

## Rules Executed AFTER Re-pricing return

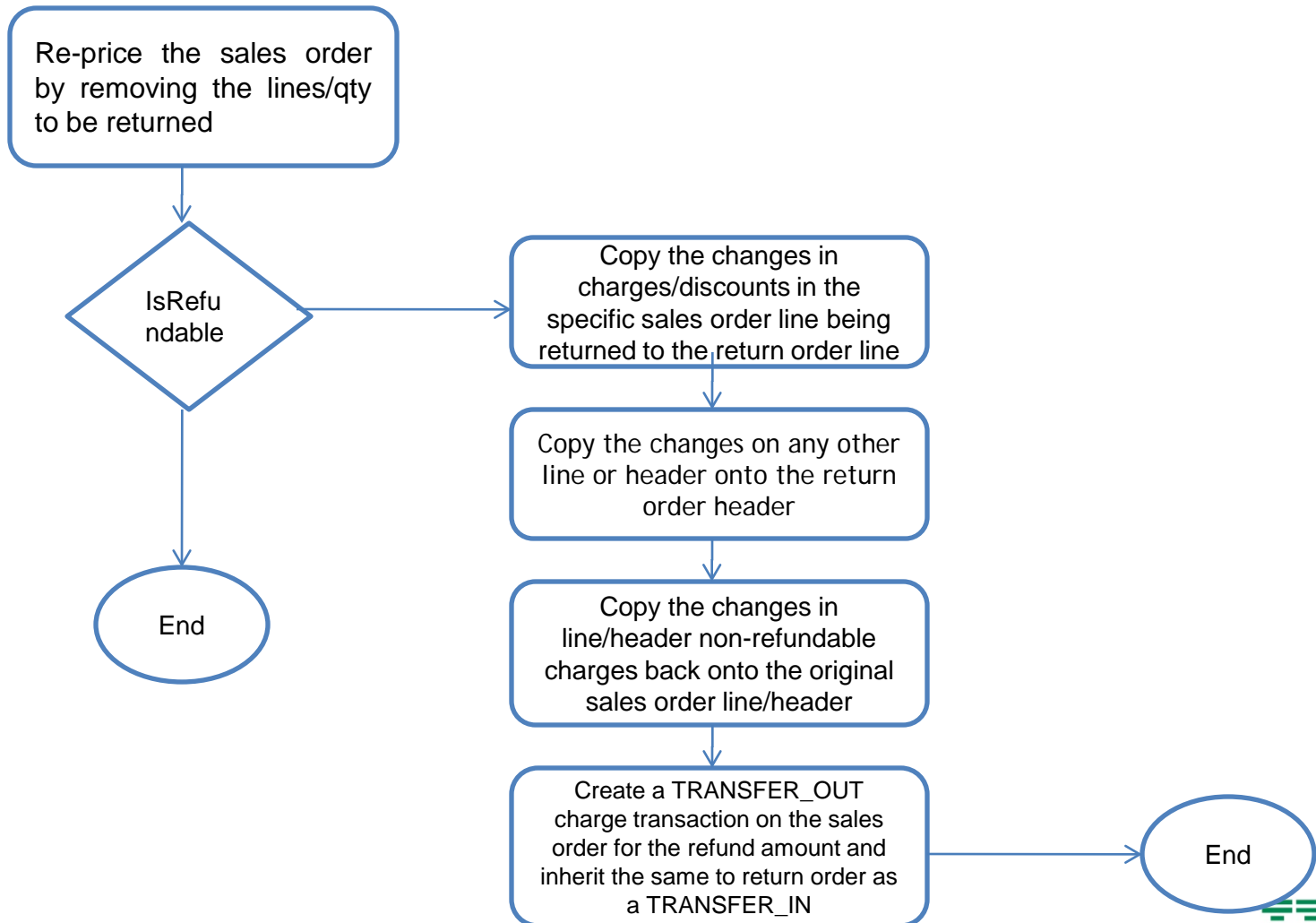
Rule Type	Description
Unit Price Limit	This validation will used to check for a unit refund value for an item on an order line.
Line Price Override Limit	The MANUAL_PRICE_OVERRIDE validation for Sales orders will be reused for the return order document type too
Blind Line	This validation is used to check whether the creation of the blind line is allowed by the enterprise.

# Re-Pricing return (1)

- The objective of the re-pricing of the return will be to compute the refund amount
- Before computing the refund, original sales order is re-priced by removing the returned lines/quantity
- The difference between line/header refundable charges between original sales order and re-priced order is copied on to return order
- Based on the AllowRepriceForReturn flag at enterprise/document type level, the re-price logic will change
- The order total on the return order will give the refund amount

# Re-Pricing return (2)

Below is the re-price return order flow for **AllowRepriceForReturn =TRUE**



# Re-Pricing return - ILLUSTRATION (1)

## Pricing Rules

1. Buy 1 PlasmaHDTV get 30% off on DVDPlayer!
2. AllowRepriceForReturn is set as "TRUE"

## Return Scenario-1

Customer want to return 1 qty of the HD TV

## Sales Order – SO1

Line #	Item Id	Unit Price	Qty	Line Price	Tax	Charges	Line Total
1	HD TV	600	2	1200	60	-60	1200
2	DVD Player	50	2	100	5	-30	75

## Line Level Charges Break-up

**Order Total 1275**

Line #	1		2	
Charge Category	Price Match	Handling	Discount	Discount
Charge Per Unit	-40	0	0	0
Charge Per Line	0	20	-30	-20
Charge Amount	-80	20	-30	-20



# Re-Pricing return - ILLUSTRATION (2)

## Return order

Line #	Item Id	Unit Price	Qty
1	HD TV	600	

Line Price reduced to account the reduction in Qty

## Re-priced sales order

Line #	Item Id	Unit Price	Qty	Line Price	Tax	Charges	Line Total
1	HD TV	600	1	600	30	-20	610
2	DVD Player	50	2	100	5	-15	90

Re-priced Order Total: 700

Price match is reduced to -40 for one qty

Handling charges are per Line, so no reduction, even if qty is reduced

Discount of DVD reduced to account reduction in Qty of HDTV

## Line Level Charges Break-up

Line #	1	2
Charge Category	Price Match	Handling
Charge Per Unit	-40	0
Charge Per Line	0	20
Charge Amount	-40	20



# Re-Pricing return - ILLUSTRATION (3)

## Return Order After re-pricing of Sales Order

Line #	Item Id	Unit Price	Qty	Line Price	Tax	Charges	Line Total
1	HD TV	600	1	600	30	-40	590

## Line Level Charges Break-up

**Refund Amount 685**

Line #	1		2	
Charge Category	Price Match	Handling	Discount	Discount
Charge Per Unit	-40	0	0	0
Charge Per Line	0	0	0	0
Charge Amount	-40	0	0	0

# Re-Pricing return - ILLUSTRATION (4)

## Pricing Rules

1. Buy 1 PlasmaHDTV get 30% off on DVDPlayer!
2. AllowRepriceForReturn is set as "FALSE"

## Return Scenario-2

Customer want to return 1 qty of the HD TV

## Sales Order – SO1

Line #	Item Id	Unit Price	Qty	Line Price	Tax	Charges	Line Total
1	HD TV	600	2	1200	60	-60	1200
2	DVD Player	50	2	100	5	-30	75

## Line Level Charges Break-up

**Order Total 1275**

Line #	1		2	
Charge Category	Price Match	Handling	Discount	Discount
Charge Per Unit	-40	0	0	0
Charge Per Line	0	20	-30	-20
Charge Amount	-80	20	-30	-20

# Re-Pricing return - ILLUSTRATION (5)

## Return order

Line #	Item Id	Unit Price	Qty
1	HD TV	600	1

## Re-priced sales order

Line #	Item Id	Unit Price	Qty	Line Price	Tax	Charges	Line Total
1	HD TV	600	1	600	30	-40	590

**Re-Priced Order Total 590**

## Line Level Charges Break-up

Line #	1		
Charge Category	Price Match	Handling	
Charge Per Unit	-40	0	
Charge Per Line	0	0	0
Charge Amount	-40	0	0

Handling charge of \$20 is not copied over since it is a flat charge per line and the order is only being returned partially

# Payment Processing Of Return Order(1)

- New API `computeRefundPayments` to set the refund amount across the payment methods
- This API is invoked by `processReturnOrder` API
- The API will set the refund amount on the new attribute `processOrderPayments` on the Payment methods
- The payments will be updated on sales order through the new event raised on request collection
- `computeRefundPayments` is stateless, if its re-run all `PlannedRefundAmounts` will be reset to zero and the refund payments will be recalculated from the available payment methods
- If `computeRefundPayments` is re-run, any payment method that has already been refunded will remain - but all other payment methods on the return will be removed before calculation

# Payment Processing Of Return Order(2)



- Previously completed refunds will be taken into consideration when computing the refund amount.

# Payment Processing Of Return Order(3)

## Case-1 : simple refund to same payment method (1)

CASH refunds to new CASH, unless > 200, then it refunds to CHECK

SVC Refunds to new SVC, unless it is < 5 then it refunds to CASH

DEBIT\_CARD refunds to new CASH

CREDIT\_CARD refunds to itself

Order	Payment Method	Amount Used	Pending Refund	Refunded	Linked Return Payment
Sales Order	CREDIT_CARD_1	100			
Return Order	CREDIT_CARD_1		100		CREDIT_CARD_1

- Returning \$100 of \$100 sales order, paid for using CREDIT\_CARD\_1
- Return order is created, and computeReturnPayments finds the CREDIT\_CARD\_1 on the sales order and adds it to the return order.
- Since it was the only payment method on the sales order and there is sufficient funds to process the return, a pending amount of \$100 is set on the return order

# Payment Processing Of Return Order(4)

## Case-1 : simple refund to same payment method (2)

After running processOrderPayments API, Refunded amount is updated and refund amount is linked to "CREDIT\_CARD\_1"

Order	Payment Method	Amount Used	Pending Refund	Refunded	Linked Return Payment
Sales Order	CREDIT_CARD_1	100			
Return Order	CREDIT_CARD_1		100	100	CREDIT_CARD_1



# Payment Processing Of Return Order(5)

## Case-2 : simple refund to new payment method (1)

Order	Payment Method	Amount Used	Pending Refund	Refunded	Linked Return Payment
Sales Order	DEBIT_CARD_1	100			
Return Order	DEBIT_CARD_1		0		DEBIT_CARD_1
Return Order	CASH_1		100		

- ❖ Returning \$100 of \$100 sales order, paid for using DEBIT\_CARD\_1
- ❖ Return order is created, and computeReturnPayments finds the DEBIT\_CARD\_1 on the sales order and adds it to the return order.
- ❖ since DEBIT\_CARD is configured to refund to cash, additionally, a CASH payment method is generated (since one does not exist on the sales order) and is added to the return order
- ❖ The pending amount is left at 0 for DEBIT\_CARD\_1, as it is not configured to refund

# Payment Processing Of Return Order(6)

## Case-2 : simple refund to new payment method (2)

- After running processOrderPayments API, CASH will not be transferred back to sales order
- Sales order will be updated to acknowledge the alternate payment refund amount
- The sales order has its refund total (in this case, it's alternate refund total) updated to acknowledge the refund, though the Linked Return Payment Method information

Order	Payment Method	Amount Used	Pending Refund	Refunded	Linked Return Payment
Sales Order	DEBIT_CARD_1	100			
Return Order	DEBIT_CARD_1		0	0	DEBIT_CARD_1
Return Order	CASH_1		100	100	DEBIT_CARD_1

# Payment Processing Of Return Order(7)

## Case-3 : Multi payment method refund (1)

Order	Payment Method	Amount Used	Pending Refund	Refunded	Linked Return Payment
Sales Order	CREDIT_CARD_1	150			
	DEBIT_CARD_1	100			
	DEBIT_CARD_2	150			
Return Order	CREDIT_CARD_1		150	150	CREDIT_CARD_1
	DEBIT_CARD_1		0	0	DEBIT_CARD_1
	DEBIT_CARD_2		0	0	DEBIT_CARD_2
	CHECK		250 (100+150)	250 (100+150)	DEBIT_CARD_1, DEBIT_CARD_2

- ❖ Returning \$400 of \$400 sales order, paid for using three payment methods
- ❖ All three original payment methods are added to the order, though only the CREDIT\_CARD\_1 has a pending refund
- ❖ Debit card 1 and 2 cannot be returned to CASH, as it exceeds \$200, so CHECK is added to return Order
- ❖ CHECK is linked to DEBIT\_CARD\_1 and DEBIT\_CARD 2

# Payment Processing Of Return Order(8)

## Case-4 : Multi payment method refund , Multiple refunds with manual overrides(1)

Order	Payment Method	Amount Used	Pending Refund	Refunded	Linked Return Payment
Sales Order	CREDIT_CARD_1	150			
	DEBIT_CARD_1	100		100	
	DEBIT_CARD_2	150		25	
Return Order	CREDIT_CARD_1		0		CREDIT_CARD_1
	DEBIT_CARD_1		0		DEBIT_CARD_1
	DEBIT_CARD_2		0		DEBIT_CARD_2
	CHECK		125(100+25)	125	DEBIT_CARD_1 and 2

- ❖ Returning \$125 of \$400 sales order, paid for using three payment methods
- ❖ All three original payment methods are added to the order, though none of them will have a pending refund amount
- ❖ A new CASH payment method will be linked to DEBIT\_CARD\_1
- ❖ After processing the refunds payments, refunded amount will be updated

# Payment Processing Of Return Order(9)

## Case-4 : Multi payment method refund , Multiple refunds with manual overrides(2)

- ❖ Second refund of \$230 requested
- ❖ DEBIT\_CARD\_1 has already been refunded, it is not eligible for further refunding
- ❖ \$25 of DEBIT\_CARD\_2 is similarly unavailable
- ❖ The refund will be to CASH, not check, as each individual refund was under \$200

Order	Payment Method	Amount Used	Pending Refund	Refunded	Linked Return Payment
Sales Order	CREDIT_CARD_1	150			
	DEBIT_CARD_1	100			
	DEBIT_CARD_2	150			
Return Order -1	CREDIT_CARD_1		0		CREDIT_CARD_1
	DEBIT_CARD_1		0		DEBIT_CARD_1
	DEBIT_CARD_2		0		DEBIT_CARD_2
	CHECK		125(100+25)	125	DEBIT_CARD_1 and 2
Return Order -2	CREDIT_CARD_1	105			CREDIT_CARD_1
	CASH	125			DEBIT_CARD_2

# Payment Processing Of Return Order(10)

## Case-4 : Multi payment method refund , Multiple refunds with manual overrides(3)

- ❖ Customer decides they want cash back for the entire transaction
- ❖ Manager agrees for the cash back for debit card and check for credit card
- ❖ No link between the CHECK and the CREDIT\_CARD, as the check is manually added.

Order	Payment Method	Amount Used	Pending Refund	Refunded	Linked Return Payment
Sales Order	CREDIT_CARD_1	150			
	DEBIT_CARD_1	100			
	DEBIT_CARD_2	150			
Return Order -1	CREDIT_CARD_1		0		CREDIT_CARD_1
	DEBIT_CARD_1		0		DEBIT_CARD_1
	DEBIT_CARD_2		0		DEBIT_CARD_2
	CHECK		125(100+25)	125	DEBIT_CARD_1 and 2
Return Order -2	CREDIT_CARD_1	0		0	CREDIT_CARD_1
	CASH	125		125	DEBIT_CARD_2
	CHECK	105		125	

# Payment Processing Of Return Order(11)

## Case-5 : Multi Sales Orders

- ❖ Returning \$250 of \$250 sales order, \$300 of \$300 sales order in same return
- ❖ The return contains the union of all payment methods from the sales orders
- ❖ The CREDIT\_CARD\_1 payment method is identified as being the same payment method and thus it is represented once in the return order but will return to each sales order.

Order	Payment Method	Amount Used	Pending Refund	Refunded	Linked Return Payment
Sales Order-1	CREDIT_CARD_1	150			
	DEBIT_CARD_1	100			
Sales Order-2	CREDIT_CARD_1	150			
	DEBIT_CARD_2	150			
Return Order	CREDIT_CARD_1		300(150+150)	300	CREDIT_CARD_1 and 2
	DEBIT_CARD_1		0		DEBIT_CARD_1
	DEBIT_CARD_2		0		DEBIT_CARD_2
	CHECK		250(100+150)		DEBIT_CARD_1 and 2

# Payment Processing Of Return Order(12)

## Case-5 : Exchange Orders(1)

- ❖ Returning \$125 of a \$250 sales order and Exchanging it for a \$100 item
- ❖ The return has the credit card on it, with a pending refund of 25
- ❖ The CREDIT\_CARD\_1 payment method is identified as being the same payment method and thus it is represented once in the return order but will return to each sales order.
- ❖ When the payment processing is run on the draft return only , only credit card refund is processed

Order	Payment Method	Amount Used	Pending Refund/ Transfer	Refunded/ Transferred	Linked Return Payment
Sales Order-1	CREDIT_CARD_1	250			
Return Order	CREDIT_CARD_1		25	25	CREDIT_CARD_1
	TRASFER IN		125		Sales Order
	TRANSFER OUT		100		Exchange Order



# Payment Processing Of Return Order(13)

## Case-5 : Exchange Orders(2)

- ❖ Return is confirmed, exchange is confirmed
- ❖ Return is invoiced, the pending transfers will execute, and will create two sets of transfer records
- ❖ A transfer in transfer out will be created between the sales and the return, as well as the return and the exchange

Order	Payment Method	Amount Used	Pending Refund/ Transfer	Refunded/ Transferred	Linked Return Payment
Sales Order-1	CREDIT_CARD_1	250		25	
	TRANSFER_OUT			100	Return Order
Return Order	CREDIT_CARD_1		25	25	CREDIT_CARD_1
	TRANSFER_IN			100	SALES_ORDER
	TRANSFER_OUT			100	EXCHANGE_ORDER
EXCHANGE_ORDER	TRABSFER_IN			100	RETURN