


## 60 Incidencias de reversas por caída de Origis

	Documento de causa raíz sobre incidentes N3		
Autor	Versión	Fecha de elaboración	Fecha autorización
	1.0	30 mar 2026	25 Mayo 2026

### 1.- Control de Versiones

Versión	Fecha	Autor	Descripción
1.0	16/04/2026	Sergio Calalpa Torres	Se reporta que para el día 29/01/2025 se identificaron más de 300 reversas entre las 12:00 a 13:00 hrs, durante este periodo se reportan caidas de ORIGIS

### 2.- Descripción general

Se reporta que para el día 29/01/2025 se identificaron más de 300 reversas entre las 12:00 a 13:00 hrs, donde en comparativa con el día anterior 28/01/2025 el número de reversas por hora es

mucho menor a lo reportado.

	hora	numReversas
1	0	1
2	7	5
3	8	7
4	9	6
5	10	9
6	11	35
7	12	335
8	13	71
9	14	26
10	15	22
11	16	23
12	17	4
13	23	5

29/01/2025

	hora	numReversas
1	0	1
2	5	1
3	6	2
4	7	1
5	8	7
6	9	14
7	10	8
8	11	19
9	12	23
10	13	15
11	14	17
12	15	15
13	16	22
14	17	22
15	18	26
16	19	40
17	20	24
18	21	24
19	22	15
20	23	6

28/01/2025

Durante este proceso se toma una venta como referencia de análisis **V30372000042893** ya que sucedió en el lapso de horas reportadas y al igual fue reversada.

(No column name)	idTransaccion	codigoMovimientoCaja	fechaTransaccion	hora	numReversas	horaTransaccionSector	codigoEmpleador	codigoDependiente	codigoCajero	codigoTenedor	idFechaOperacion	fechaOperacion	ipOperacion
1	FFDC9A5-72DE-EP11-8287-80CF3784043	279895D-72DE-EP11-8287-80CF3784043	V3037200042893	2025-01-29 12:56:01.217	C	28418172	78.4827	NALL	2025-01-29	2025-01-29	2025-01-29	2025-01-29	C

(No column name)	idTransaccion	region	codigoProducto	codigoRelacionado	unidad	precioRegular	precioVenta	precioMaximo	cantidadTendita	tesaha	importeA	importeDescuento	importeCajero	codigoDepartamento	importeComision	snDevolucion	ventasInventencia	costoUnitario	porcentajeCajero	fechaReferencia	estado
1	FFDC9A5-72DE-EP11-8287-80CF3784043	1	1549114104	7501871720644	0	543.0000	481.0000	740.4400	1.0000	0.0000	0.0000	62.0000	0.0000	7	0.0000	0	0.0000	0.0000	0.0000	NALL	0
2	FFDC9A5-72DE-EP11-8287-80CF3784043	2	694761955	7501088004365	0	979.0000	979.0000	1290.7000	1.0000	0.0000	0.0000	0.0000	0.0000	7	0.0000	0	0.0000	0.0000	0.0000	NALL	0
3	FFDC9A5-72DE-EP11-8287-80CF3784043	3	1295642658	7501304057160	0	648.0000	648.0000	888.0000	1.0000	0.0000	0.0000	0.0000	0.0000	7	0.0000	0	0.0000	0.0000	0.0000	NALL	0
4	FFDC9A5-72DE-EP11-8287-80CF3784043	4	1919226626	3337875631888	0	544.8279	435.7759	719.2241	1.0000	16.0000	69.7241	109.2516	0.0000	2	0.0000	0	0.0000	0.0000	0.0000	NALL	0
5	FFDC9A5-72DE-EP11-8287-80CF3784043	5	495239011	7501128112802	0	42.2413	57.0258	1.0000	16.0000	6.7588	0.0000	10.0000	0.0000	2	0.0000	0	0.0000	0.0000	0.0000	NALL	0
6	FFDC9A5-72DE-EP11-8287-80CF3784043	6	1064434765	7501825302056	0	65.0000	605.4100	1.0000	0.0000	0.0000	10.0000	0.0000	0.0000	7	0.0000	0	0.0000	0.0000	0.0000	NALL	0

(No column name)	codigoMovimientoCaja	codigoTurno	tipoMovimiento	idFechaOperacion	codigoBursatil	fechaOperacion	horaOperacion	horaCierre	estadoMovEntrenamiento	tipoTransaccion	codigoUsuario	fechaActualizacion	syncCreadoFecha	syncActualizadoFecha	syncActualizadoPor	syncVersionRegistro	syncTransferido	codigoMoneda	importePagoMonedaEstimada
1	279895D-72DE-EP11-8287-80CF3784043	13019378-30DE-EP11-8287-80CF3784043	1	20250129003037	3037	2025-01-29	V30372000042893	C7303720000990	0	V	4633	2025-01-29 12:53:45.527	2025-01-29 12:54:58.527	2025-01-29 14:06:44.083	FARMAX S'NC LP				

(No column name)	codigoMovimientoCaja	codigoFormaPago	importePago	importeComisionBancaria	importeIVA	autorizacion	documento	estadoFormaPago	syncCreadoFecha	syncActualizadoFecha	syncActualizadoPor	syncVersionRegistro	syncTransferido			
1	13019378-30DE-EP11-8287-80CF3784043	3037	21258	0	20250129003037	2025-01-29	750.0000	2025-01-29 14:06:32.08.140	2025-01-29 14:06:18.130	1	4633	2025-01-29 14:06:18.130	1000560701	2	2025-01-29 13:33:43.830	1

(No column name)	codigoBursatil	fechaTransaccion	idFormaPago	importe	importeTajeta	status	codigoFormaPago	idReferencia	estadoOperacion	mensajeDetalle	codigoRespuesta	fechaTransaccion	codigoFormaPago	codigoFormaPagoElectronico	codigoUsuario	fechaActualizacion	fechaTransaccionAplic
1	3037	20250129	V30372000042893	2171.5000	4134960000003975	1	11	0	3	00	00	2025-01-29 12:54:18.000	0000020042893	P	DEB37879-72DE-EP11-8287-80CF3784043	4633	2025-01-29 12:54:00
2	3037	20250129	V30372000042893	2171.5000	4134960000003975	1	11	0	0	00	00	2025-01-29 12:54:59.233	0000020042893	R	C6886509-72DE-EP11-8287-80CF3784043	4633	2025-01-29 12:54:59.313

(No column name)	idTransaccion	region	consecutivo	codigoPromocion	codigoRedPromocion	codigoCondicionPromocion	secuencialPromocion	asistencia	cantidad	metodoCobro	numeroPunto	tipoExhibicion	fechaPromocion	estado	numeroAutorizacion	idSession	horaCierre	syncCreadoFecha	syncActualizadoFecha	syncActualizadoPor	syncVersionRegistro
------------------	---------------	--------	-------------	-----------------	--------------------	--------------------------	---------------------	------------	----------	-------------	-------------	----------------	----------------	--------	--------------------	-----------	------------	-----------------	----------------------	--------------------	---------------------

La transacción se encuentra cancelada, a consecuencia del pago reversado, el cual se observa en la tabla genFormasPagosElectronicas, solo que no se observa un código de respuesta diferente de

0.

Después de que ocurrió la reversa se volvió a intentar el cobro con el folio **V30372000042894** terminando exitoso.

codigoSucursal	folioVenta	codigoFomaPago	status	fechaTransaccion	monto	numeroTarjeta	esReversa	mesesDiferido	codigoRespuesta	numeroPagare
3037	V30372000042894	11	1	2025-01-29 12:56:13.000	2717.5000	41340600XXXX5975	0	3	00	110216239044
3037	V30372000042893	11	1	2025-01-29 12:54:59.233	2717.5000	41340600XXXX5975	1	0	00	110216238800
3037	V30372000042893	11	1	2025-01-29 12:54:18.000	2717.5000	41340600XXXX5975	0	3	00	110216238698

La cuestión es, porque si no se observa un código de respuesta diferente de 0 en la tabla **genFormasPagosElectronicas** se realizó la reversa.

### 3.- Diagnóstico

Durante la revisión del caso se revisó el log FARMAX\_POS\_PINPADLD, en donde se observa la cancelación de la autorización **386428**.

```
[2025-01-29 12:54:05:569],FARMAX_POS,PINPADLD,INFO,,Revisando conexión: IP 192.168.152.17 Puerto 7000
[2025-01-29 12:54:18:220],FARMAX_POS,PINPADLD,INFO,,Iniciando llamado con parametros: IP - 192.168.152.17, Puerto - 7000
[2025-01-29 12:54:22:616],FARMAX_POS,PINPADLD,INFO,,EnviaPeticionVenta Autorizado - Venta: 42893, Monto: 2717.50, CardType: 12
[2025-01-29 12:54:52:971],FARMAX_POS,PINPADLD,INFO,,Iniciando llamado con parametros: IP - 192.168.152.17, Puerto - 7000
[2025-01-29 12:54:53:120],FARMAX_POS,PINPADLD,INFO,,cancelacionAutorizacion Autorizado - Venta: 42893, Monto: 2717.50, OriginalTransaction: 42893, Autorización: 386428
[2025-01-29 12:55:49:421],FARMAX_POS,PINPADLD,INFO,,DispositivoPresente: True
```

En el log FARMAX\_TILL\_SERVER\_SALE se observa el siguiente error:

```
[2025-01-29
12:54:49:380],FARMAX_TILL_SERVER,SALE,ERROR,[COMPLETEPAYMENTSALE],SaleManager
CompletePaymentSale. Error en CommitLoyaltyPrograms. Folio: V30372000042893
```

En el log FARMAX\_TILL\_SERVER\_LOYALTYPROGRAM se observa el siguiente error con el código **LPF001**:

[2025-01-29  
12:54:49:378],FARMAX\_TILL\_SERVER,LOYALTYPROGRAM,ERROR,[COMMIT],LoyaltySaleManage  
r - Commit. Retorno error LoyaltyProgramServiceClient.CommitLoyaltyRewards  
[2025-01-29  
12:54:49:378],FARMAX\_TILL\_SERVER,LOYALTYPROGRAM,ERROR,[COMMIT],LoyaltySaleManage  
r - Commit. **Codigo:LPF001**  
Mensaje:FAhorro.POS.Service.POSLog.DataContracts.POSLog.DescriptionCommonData  
[2025-01-29  
12:54:49:379],FARMAX\_TILL\_SERVER,LOYALTYPROGRAM,ERROR,[COMMIT],LoyaltySaleManage  
r - Commit. Retorno error LoyaltyProgramServiceClient.CommitLoyaltyRewards 2

El código obtenido **LPF001** es revisado en código y podemos observar que corresponde a un error de conexión con **ORIGIS**.

```
> /// Contiene las constantes de los Errores del Negocio ...  
Juan Murillo il code  
public static class ErrorCode  
{  
> /// No se encontró el Producto ...  
public const string NonexistentTemplate = "IMP001";  
> /// Indica un error en el ambito en que se intento realizar la operación ...  
public const string InvalidScope = "IS9991";  
> /// Indica que la venta fue cancelada por el servicio de transacciones. ...  
public const string SaleWasCanceled = "IS9992";  
> /// Indica un error de conexión de red. ...  
public const string AlreadyReversed = "LPF001";
```

Por lo cual se procede a revisar el archivo log de los servidores centrales FARMAX\_WSC\_SERVER\_LOYALTYPROGRAM.

Detectando un error de red en el servicio de **ORIGIS**, ya que no estaba disponible y corresponde al horario en el que la transacción fue efectuada.

[2025-01-29  
12:54:22:352],FARMAX\_WSC,LOYALTYPROGRAM,ERROR,[BACKGROUNDEXECUTIONTASK],OriginsManager - Commit - Exception: The HTTP request to 'https://pos.monederodelahorro.net/online3/ctransaccion.asmx' has exceeded the allotted timeout of 00:00:25. The time allotted to this operation may have been a portion of a longer timeout.

[2025-01-29  
12:54:22:352],FARMAX\_WSC,LOYALTYPROGRAM,ERROR,[BACKGROUNDEXECUTIONTASK],InnerException: The request was aborted: The request was canceled.

[2025-01-29  
12:54:22:352],FARMAX\_WSC,LOYALTYPROGRAM,ERROR,[BACKGROUNDEXECUTIONTASK],Stack Trace: at System.Runtime.AsyncResult.End[TAsyncResult](IAsyncResult result)  
at System.ServiceModel.Channels.ServiceChannel.SendAsyncResult.End(SendAsyncResult result)  
at System.ServiceModel.Channels.ServiceChannel.EndCall(String action, Object[] outs, IAsyncResult result)  
at  
System.ServiceModel.Channels.ServiceChannelProxy.TaskCreator.<>c\_\_DisplayClass7\_0`1.<CreateGenericTask>b\_\_0(IAsyncResult asyncResult)  
at System.Threading.Tasks.TaskFactory`1.FromAsyncCoreLogic(IAsyncResult iar, Func`2 endFunction, Action`1 endAction, Task`1 promise, Boolean requiresSynchronization)  
--- End of stack trace from previous location where exception was thrown ---  
at System.Runtime.CompilerServices.TaskAwaiter.ThrowForNonSuccess(Task task)  
at  
System.Runtime.CompilerServices.TaskAwaiter.HandleNonSuccessAndDebuggerNotification(Task task)  
at System.Runtime.CompilerServices.TaskAwaiter.ValidateEnd(Task task)  
at System.Runtime.CompilerServices.TaskAwaiter`1.GetResult()

[2025-01-29  
12:55:21:249],FARMAX\_WSC,LOYALTYPROGRAM,ERROR,[FLUSH],GetCardContactDataLookup - Exception: El servicio de Origis no está disponible.  
[2025-01-29 12:55:21:249],FARMAX\_WSC,LOYALTYPROGRAM,ERROR,[FLUSH],InnerException: No detail  
[2025-01-29 12:55:21:249],FARMAX\_WSC,LOYALTYPROGRAM,ERROR,[FLUSH],Stack Trace: at FAhorro.POS.BusinessLogic.LoyaltyProgram.LoyaltyProgramOrigisPosMethods.<GetAfiliationContactSearchi ng>d\_\_27.MoveNext() in C:\builds-or\projects\q9WN7bzD\2\farmaciasdelahorro\coe\farmax\pos-or\pos\Farmax\Src\POS\BusinessLogic.LoyaltyProgram\LoyaltyProgramOrigisPosMethods.cs:line 286  
--- End of stack trace from previous location where exception was thrown ---  
at System.Runtime.ExceptionServices.ExceptionDispatchInfo.Throw()  
at System.Runtime.CompilerServices.TaskAwaiter.ThrowForNonSuccess(Task task)  
at  
System.Runtime.CompilerServices.TaskAwaiter.HandleNonSuccessAndDebuggerNotification(Task task)  
at System.Runtime.CompilerServices.TaskAwaiter.ValidateEnd(Task task)  
at System.Runtime.CompilerServices.TaskAwaiter`1.GetResult()  
at  
FAhorro.POS.BusinessLogic.LoyaltyProgram.LoyaltyProgramManager.<GetCardContactDataLookup>d\_\_17. MoveNext() in

El servicio de ORIGS estuvo inestable desde el horario **[2025-01-29 12:16:46:427]** a **[2025-01-29 13:08:48:551]** lo cual coincide con las horas en las que se reporta el problema.

El día de hoy nos reportaron un evento en nuestras transacciones con la forma de pago de tarjetas bancarias revisamos y detectamos que en el periodo aproximado de **12:17 a 13:09** es donde se producen el mayor número de reversas de transacciones, es decir se aplicaron Debit reversal y Credit reversal

Ejemplo de una reversa

e #	Store Name	Date-Time	Transaction Type	Amount	Cash Back	Response	ID Type	ID Number	STAN	Terminal	Auth. Time(ms)	Customer Name	CID
110216239044	F A COCO CORREOS	01/29/2025 12:56:16.700	0200 - Credit Sale	\$ 2,717.50		00 - Approved	Visa	413406*****5975	42894	20	823		80
110216238800	F A COCO CORREOS	01/29/2025 12:54:55.870	0400 - Debit Reversal	\$ 2,717.50		00 - Approved	Visa	413406*****5975	42893	20	90		80
110216238698	F A COCO CORREOS	01/29/2025 12:54:21.743	0200 - Credit Sale	\$ 2,717.50		00 - Approved	Visa	413406*****5975	42893	20	724		80

En farmax se tienen igual dichas transacciones, pero sin código de error

codigoSucursal	folioVenta	codigoFormaPago	status	fechaTransaccion	monto	numeroTarjeta	esReversa	mesesDiferido	codigoRespuesta	numeroPagare
3037	V30372000042894	11	1	2025-01-29 12:56:13.000	2717.5000	4134060000005975	0	3	00	110216239044
3037	V30372000042893	11	1	2025-01-29 12:54:59.233	2717.5000	4134060000005975	1	0	00	110216238800
3037	V30372000042893	11	1	2025-01-29 12:54:18.000	2717.5000	4134060000005975	0	3	00	110216238698

Lo que nos gustaría saber es como es el flujo de la transacción, ya que la primera operación fluyo y tuvo una respuesta exitosa y con numero de autorización, pero unos segundos despues se lanza la reversa. Que detona para que se lance dicha reversa?

Posterior se vuelve a aplicar y ya se queda exitosa nuevamente

Como este ejemplo hay muchos el día de hoy en el periodo indicado, le agrego una gráfica en que horarios se tuvo la mayor concentración

<b>Cronología Farmax</b>	
Comienza la transacción	2025-01-29 <b>12:53:32:562</b>
Error - <b>CommitLoyaltyPrograms</b>	2025-01-29 <b>12:54:49:380</b>
Transacción cancelada	2025-01-29 <b>12:55:01:400</b>

<b>Cronología Monedero</b> (según Logs centrales)	
getAfiliacionPerfil	2025-01-29 12:52:13:275
setInicioTrans2	2025-01-29 12:52:13:311
getTarjetaDatos	2025-01-29 12:52:13:311
getPuntosProductos	2025-01-29 12:53:36:885
getTransStatus	2025-01-29 12:54:51:792

---

#### 4.- Replicación del caso

Para realizar la replicación del caso se provoca una desconexión durante el proceso de pago, dejando que al inicio de la venta se pueda iniciar una sesión de Monedero, pero al final del pago provocar la desconexión.

Se realiza la selección del producto para la venta en cuestión e ingresamos el monedero.



Para este ejemplo realizamos el pago con Tarjeta de Crédito ACI (AC).

Formas de Pago

E EFECTIVO (CA)	0.00	N MONEDERO VIRTUAL	0.00	W Cash Payment (Medio de Pago)	0.00
<b>T TARJETA DE CREDITO ACI (AC)</b>	<b>7.50</b>	P PUNTOS BBVA	0.00	P CLARORAY	0.00
F TARJETA DE DEBITO ACI (ADI)	0.00	G TARJETA GIFT CARD	0.00	Q VALES	0.00
X TARJETA DE CREDITO ACI AMEX	0.00	K TARJETA T-ENVIO	0.00	I TERMINAL TPV	0.00
O MONEDERO ELECTRONICO	0.00	C Pago CODI	0.00		

Subtotal: **\$7.50**      I.V.A: **\$0.00**      Total: **\$7.50**

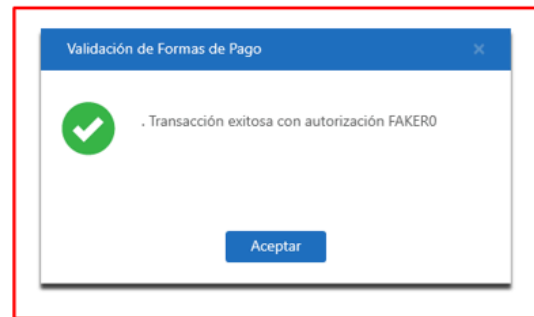
Tu Donativo **\$0.00**      Tu Ahorro **\$0.00**      Por Pagar **\$0.00**

←

Realizamos el pago y esperamos a tener la confirmación del pago por Tarjeta de Crédito ACI (AC).

## Formas de Pago

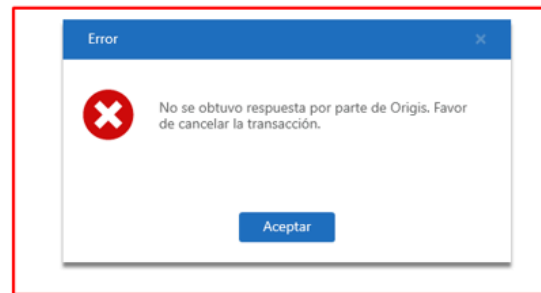
<input type="checkbox"/> E	EFFECTIVO (CA)	<input type="text" value="0.00"/>	<input type="checkbox"/> N	MONEDERO VIRTUAL	<input type="text" value="0.00"/>	<input type="checkbox"/> W	Cash Payment (Medio de Pago)	<input type="text" value="0.00"/>
<input type="checkbox"/> T	TARJETA DE CREDITO ACI (AC)	<input type="text" value="7.50"/>	<input type="checkbox"/>	PUNTOS BBVA	<input type="text" value="0.00"/>	<input type="checkbox"/> P	CLAROPAY	<input type="text" value="0.00"/>
<input type="checkbox"/> F	TARJETA DE DEBITO ACI (AD)	<input type="text" value="0.00"/>	<input type="checkbox"/> G	TARJETA GIFT CARD	<input type="text" value="0.00"/>	<input type="checkbox"/> Q	VALES	<input type="text" value="0.00"/>
<input type="checkbox"/> X	TARJETA DE CREDITO ACI AMEX	<input type="text" value="0.00"/>	<input type="checkbox"/> K	TARJETA T-ENVIO	<input type="text" value="0.00"/>	<input type="checkbox"/> I	TERMINAL TPV	<input type="text" value="0.00"/>
<input type="checkbox"/> O	MONEDERO ELECTRONICO	<input type="text" value="0.00"/>	<input type="checkbox"/> C	Pago CODI	<input type="text" value="0.00"/>			



Generamos la desconexión del servicio de monedero lo que provoca que se realice la cancelación total de la venta.

## Formas de Pago

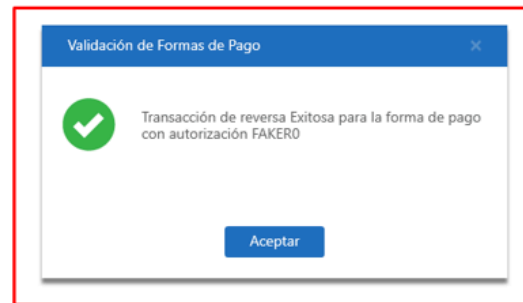
<input type="checkbox"/> E	EFFECTIVO (CA)	<input type="text" value="0.00"/>	<input type="checkbox"/> N	MONEDERO VIRTUAL	<input type="text" value="0.00"/>	<input type="checkbox"/> W	Cash Payment (Medio de Pago)	<input type="text" value="0.00"/>
<input type="checkbox"/> T	TARJETA DE CREDITO ACI (AC)	<input type="text" value="7.50"/>	<input type="checkbox"/>	PUNTOS BBVA	<input type="text" value="0.00"/>	<input type="checkbox"/> P	CLAROPAY	<input type="text" value="0.00"/>
<input type="checkbox"/> F	TARJETA DE DEBITO ACI (AD)	<input type="text" value="0.00"/>	<input type="checkbox"/> G	TARJETA GIFT CARD	<input type="text" value="0.00"/>	<input type="checkbox"/> Q	VALES	<input type="text" value="0.00"/>
<input type="checkbox"/> X	TARJETA DE CREDITO ACI AMEX	<input type="text" value="0.00"/>	<input type="checkbox"/> K	TARJETA T-ENVIO	<input type="text" value="0.00"/>	<input type="checkbox"/> I	TERMINAL TPV	<input type="text" value="0.00"/>
<input type="checkbox"/> O	MONEDERO ELECTRONICO	<input type="text" value="0.00"/>	<input type="checkbox"/> C	Pago CODI	<input type="text" value="0.00"/>			



Si damos click sobre el botón de Aceptar se realiza la reversa de toda la venta.

Formas de Pago

<input type="checkbox"/> E	EFFECTIVO (CA)	<input type="text" value="0.00"/>	<input type="checkbox"/> N	MONEDERO VIRTUAL	<input type="text" value="0.00"/>	<input type="checkbox"/> W	Cash Payment (Medio de Pago)	<input type="text" value="0.00"/>
<input type="checkbox"/> T	TARJETA DE CREDITO ACI (AC)	<input type="text" value="7.50"/>	<input type="checkbox"/>	PUNTOS BBVA	<input type="text" value="0.00"/>	<input type="checkbox"/> P	CLAROPAY	<input type="text" value="0.00"/>
<input type="checkbox"/> F	TARJETA DE DEBITO ACI (AD)	<input type="text" value="0.00"/>	<input type="checkbox"/> G	TARJETA GIFT CARD	<input type="text" value="0.00"/>	<input type="checkbox"/> Q	VALES	<input type="text" value="0.00"/>
<input type="checkbox"/> X	TARJETA DE CREDITO ACI AMEX	<input type="text" value="0.00"/>	<input type="checkbox"/> K	TARJETA T-ENVIO	<input type="text" value="0.00"/>	<input type="checkbox"/> I	TERMINAL TPV	<input type="text" value="0.00"/>
<input type="checkbox"/> O	MONEDERO ELECTRONICO	<input type="text" value="0.00"/>	<input type="checkbox"/> C	Pago CODI	<input type="text" value="0.00"/>			



COMERCIALIZADORA FARMACEUTICA DE CHIAPAS SAPI DE CV  
TLAXCALA 151 PISO 4 OFNA 407 HIPODROMO  
06100 MEXICO CUAUHTEMOC CIUDAD DE MEXICO  
RFC: CFC110121742  
No. Tarjeta/Cuenta: XXXXXXXXXXXXX4025  
22/04/26 22:00:17  
00 - Transacción de reversa Exitosa para la forma de  
pago con autorización FAKERO

## 5.- Causa del problema

Se realiza una revisión de los textos internos que nos dan los logs reportados y al obtener la replicación del caso detectamos el mismo patrón de mensajes, lo cual nos indica que este comportamiento de intermitencia en el servicio nos puede ocasionar una reversa completa de la venta a pesar de tener el pago autorizado de nuestra venta.

```
20260420_FARMAX_TILL_SERVER_LOYALTYPROGRAM.XLOG:1: [2026-04-20 15:03:26:071], FARMAX_TILL_SERVER, LOYALTYPROGRAM, ERROR, [WRITEERROR], LoyaltySaleManager - Commit. Return  
o error LoyaltyProgramServiceClient.CommitLoyaltyRewards  
20260420_FARMAX_TILL_SERVER_LOYALTYPROGRAM.XLOG:2: [2026-04-20 15:03:26:081], FARMAX_TILL_SERVER, LOYALTYPROGRAM, ERROR, [WRITEERROR], LoyaltySaleManager - Commit. Código  
:LPF002 Mensaje:Fáhorro.POS.Service.POSLog.DataContracts.POSLog.DescriptionCommonData  
20260420_FARMAX_TILL_SERVER_LOYALTYPROGRAM.XLOG:3: [2026-04-20 15:03:26:084], FARMAX_TILL_SERVER, LOYALTYPROGRAM, ERROR, [WRITEERROR], LoyaltySaleManager - Commit. Return  
o error LoyaltyProgramServiceClient.CommitLoyaltyRewards 2
```

Se realiza la revisión en código para este momento de los registros en logs pudiendo notar que los siguientes mensajes de error pueden realizar este efecto se reversa completa de la venta.

- LPF001 (AlreadyReversed)
- LPF002 (ReverseError)
- LP012 (ConnectionError)
- LP011 (UnexpectedError)

```

OrigisManager.cs
/// <summary>
/// Finaliza la sesión con el tercero e indica que las bonificaciones indicadas fueron entregadas
/// </summary>
/// <param name="benefits">Información de la tarjeta del cual se consultaran los beneficios</param>
/// <returns>Regresa la información de la tarjeta validada en el objeto <see cref="LoyaltyTransaction"/></returns>
[1+1] Usages Juan Murillo +3
public override async Task<ResultStruct<LoyaltyBenefits>> Commit(LoyaltyTransaction benefits)
{
    try
    {
        using (var serviceClient = new Monedero_del_AhorroSoapClient(endpointConfigurationName: "Monedero_del_AhorroSoap", checkConnection: false, enableOrigisLog: true))
        {
            var accountResult = new LoyaltyBenefits();

            // se verifica si la transaccion ya fue procesada, si ya fue procesada se regresa el resultado obtenido por proveedor
            var checkIfExists :ResultStruct<List<...>> = this.GetAproveLogLoyaltyProgram(new CancelTokens { BusinessId = Convert.ToInt32(benefits.Branch), tillId =
            Convert.ToInt32(benefits.TillId), TransactionId = new[] { benefits.TransactionId } });
            if (checkIfExists.Result.Any(t :genProgramaLealtadBifacoraReg => t.tipoPrograma == this.ConvertEnumLoyaltyToDBLoyalty(benefits.LoyaltyProgramID))) {...}
            else {...}

            return new ResultStruct<LoyaltyBenefits>(accountResult);
        }
    }
    catch (Exception ex)
    {
        Log.WriteException(SubModule, ex, methodName: "OriginsManager - Commit");
        if (ex is CommunicationException || ex is TimeoutException)
        {
            //BLINDAJEMDA
            try
            {
                var loyaltyBenefits = new LoyaltyBenefits() { Card = benefits.Card, Branch = benefits.Branch, TillId = benefits.TillId, PayrollID = benefits
                .PayrollID, SessionId = benefits.SessionId, ShoppingBasket = benefits.ShoppingBasket, FolioTransaction = benefits.FolioTransaction, UserId =
                benefits.UserId, OperationDate = benefits.OperationDate, TransactionId = benefits.TransactionId };
                var result = this.GetTransactionStatus(loyaltyBenefits).Result;
                if (!string.IsNullOrEmpty(result.Result.AuthorizationCode) && result.Result.AuthorizationCode != "0")
                {
                    loyaltyBenefits.AuthorizationCode = result.Result.AuthorizationCode;
                    if (!this.ReverseTransaction(loyaltyBenefits).Result.HasError)
                    {
                        return new ResultStruct<LoyaltyBenefits>(ErrorCode.AlreadyReversed, errorDescription: LoyaltyProgramConstant.OrigisTimeout);
                    }
                }
                else
                {
                    return new ResultStruct<LoyaltyBenefits>(ErrorCode.AlreadyReversed, errorDescription: $"{result.Result.ErrorId} - {result.Result.Message}
                    {Environment.NewLine}{LoyaltyProgramConstant.OrigisGetTransError}");
                }
            }
            catch (Exception exe)
            {
                Log.WriteException(SubModule, exe, methodName: "OriginsManager-GetTrans-Reverse");
                return new ResultStruct<LoyaltyBenefits>(ErrorCode.ReverseError, errorDescription: LoyaltyProgramConstant.OrigisTimeout);
            }

            return new ResultStruct<LoyaltyBenefits>(ErrorCode.ConnectionError, errorDescription: LoyaltyProgramConstant.OrigisNotAvaliableMsg);
        }

        return new ResultStruct<LoyaltyBenefits>(ErrorCode.UnexpectedError, errorDescription: ex.Message);
    }
}

```

---

La causa de las reversas se debe a la intermitencia que se tuvo del servicio de **ORIGIS** durante el día en el periodo de **[2025-01-29 12:16:46:427]** a **[2025-01-29 13:08:48:551]** lo cual dentro del código esta establecido como una reversa completa de nuestra venta para poder ser intentada después nuevamente.

De este modo al realizar un segundo intento y tener una respuesta correcta de **ORIGIS** se podía concluir la venta de manera normal.

#### 6.- Propuesta de solución

Este caso aplica a solucionarse con otro fix ya existente: **89 Data Fixes - MDA Puntos de monedero son generados y guardados en la venta, aunque ocurrió error en respuesta de servicio de Origis**, dentro del cual se soluciona la reversa completa de la venta mandándola a encolar como ventas pendientes offline.

Dentro de este punto se recibe un error similar al timeout pero para este caso es un error dentro del response lo cual nos lleva al mismo método mencionado en el diagnostico del caso donde se regresa el código de error obtenido, para realizar la acción correspondiente que para la propuesta de solución comentada en este caso es agregar los registros en **opColaOperaciones** para ser procesados posteriormente.

Para el método Commit ubicado en OrigisManager.cs se propone agregar dos ramas de catch de un try asociado a una opción de comunicación y siga su flujo para retornar el código **LP012**.

Comparativa del cambio obtenido de la rama

GET\_N3\_89\_DataFixes\_PuntosGenerados\_Con\_Error\_Origis

Commit: ad938086a4cb8e4d12ada7abc0c9a3903a0149d4

```
Farmax\Src\POS\BusinessLogic\LoyaltyProgram\Origis\OrigisManager.cs
632 return new ResultStruct<LoyaltyBenefits>(ErrorCode.AlreadyReversed, $"(result.Result.ErrorMessage)
633 (Environment.NewLine)LoyaltyProgramConstant.OrigisGetTransError");
634 }
635 catch (Exception e) when (e is CommunicationException || e is TimeoutException)
636 {
637     // Se registra el fallo de comunicacion/timeout al consultar estatus a reversar.
638     Log.WriteException(SubModule, e, "OriginsManager-GetTrans-Reverse");
639 }
640 catch (Exception e)
641 {
642     // Se registra la excepcion no controlada y se devuelve error de reversa al proceso de origen.
643     Log.WriteException(SubModule, e, "OriginsManager-GetTrans-Reverse");
644 }
645 return new ResultStruct<LoyaltyBenefits>(ErrorCode.ReverseError, LoyaltyProgramConstant.OrigisTimeout);
646 }
647 }

Farmax\Src\POS\BusinessLogic\LoyaltyProgram\Origis\OrigisManager.cs
632 return new ResultStruct<LoyaltyBenefits>(ErrorCode.AlreadyReversed, $"(result.Result.ErrorMessage)
633 (Environment.NewLine)LoyaltyProgramConstant.OrigisGetTransError");
634 }
635 catch (Exception e) when (
636     (x is CommunicationException || x is TimeoutException)
637     || (x is AggregateException agg && agg.InnerExceptions != null
638         && agg.InnerExceptions.Any(y => y is CommunicationException || y is TimeoutException)
639     )
640 )
641 {
642     // Se registra el fallo de comunicacion/timeout al consultar estatus a reversar.
643     Log.WriteException(SubModule, e, "OriginsManager-GetTrans-Reverse");
644 }
645 catch (Exception e)
646 {
647     // Se registra la excepcion no controlada y se devuelve error de reversa al proceso de origen.
648     Log.WriteException(SubModule, e, "OriginsManager-GetTrans-Reverse");
649 }
650 return new ResultStruct<LoyaltyBenefits>(ErrorCode.ReverseError, LoyaltyProgramConstant.OrigisTimeout);
651 }
652 }
```

Patch obtenido de la rama GET\_N3\_89\_DataFixes\_PuntosGenerados\_Con\_Error\_Origis

Commit: ad938086a4cb8e4d12ada7abc0c9a3903a0149d4

```
1 From ad938086a4cb8e4d12ada7abc0c9a3903a0149d4 Mon Sep 17 00:00:00 2001
2 From: Cristhian CH <cristhian.carrasco@getechnologies.com.mx>
3 Date: Tue, 24 Mar 2026 09:51:36 -0600
4 Subject: [PATCH] Fix-89-catch when CommunicationException || TimeoutException
5
6 ---
7 ../OrigisManager.cs | 13 ++++++----
8 1 file changed, 9 insertions(+), 4 deletions(-)
9
10 diff --git a/Farmax/Src/POS/BusinessLogic.LoyaltyProgram.Origis/OrigisManager.cs b/Farmax/Src/
11 index d398ecb00..bdab5aade 100644
12 --- a/Farmax/Src/POS/BusinessLogic.LoyaltyProgram.Origis/OrigisManager.cs
13 +++ b/Farmax/Src/POS/BusinessLogic.LoyaltyProgram.Origis/OrigisManager.cs
14 @@ -632,15 +632,20 @@ namespace FAhorro.POS.BusinessLogic.LoyaltyProgram.Origis
15         return new ResultStruct<LoyaltyBenefits>(ErrorCode.AlreadyReverse
16     }
17 }
18 - catch (Exception e) when (e is CommunicationException || e is TimeoutExce
19 + catch (Exception x) when (
20 + (x is CommunicationException || x is TimeoutException)
21 + || (x is AggregateException agg && agg.InnerExceptions != null
22 +     && agg.InnerExceptions.Any(y => y is CommunicationException || y is T
23 + )
24 + )
25 + {
26     // Se registra el fallo de comunicaci3n/timeout al consultar estatus
27     Log.WriteException(SubModule, e, "OriginsManager-GetTrans-Reverse");
28 +     Log.WriteException(SubModule, x, "OriginsManager-GetTrans-Reverse");
29 }
30 - catch (Exception e)
31 + catch (Exception y)
32 + {
33     // Se registra la excepci3n no controlada y se devuelve error de reve
34 -     Log.WriteException(SubModule, e, "OriginsManager-GetTrans-Reverse");
```

```
35 +         Log.WriteException(SubModule, y, "OriginsManager-GetTrans-Reverse");
36         return new ResultStruct<LoyaltyBenefits>(ErrorCode.ReverseError, Loya
37     }
38
39 --
40 GitLab
41
42
```

---

## 7.- Impacto

Se genera un impacto en la forma de manejo de errores para su reprocesamiento como una venta offline, pudiendo de esta forma concluir con una venta si se tiene alguna intermitencia o desconexión del servicio de **ORIGIS** encolando la venta.

Por otro lado, esto genera un impacto en procesamiento de encolado para concluir la venta ya que al tener un tema como este en el que fueron múltiples ventas las que se tuvieron durante el periodo de intermitencia de **ORIGIS** genera un volumen fuerte para este proceso es por esto que se realiza un mecanismo para control de reintentos y estos no se extiendan por n reintentos, pudiendo así determinar el último intento del procesamiento.

---

## 8.- Conclusión

La desconexión o intermitencia (timeouts) de **ORIGIS** durante el proceso de pago de la venta nos está generando una reversa completa de la venta, es por esto que se propone la opción de terminar estas ventas bajo el esquema offline, lo que provoca que se pueda concluir al momento la venta y posterior a esto por medio del store services concluir con la venta de manera offline.

Dentro de la solución propuesta en el fix **89 Data Fixes - MDA Puntos de monedero son generados y guardados en la venta, aunque ocurrió error en respuesta de servicio de ORIGIS**

se aborda.

- El manejo de errores obtenidos por **ORIGIS** para su correcta interpretación y de ahí continuar con el esquema offline.
- Implementación de mecanismo de control de reintentos que permita determinar el último intento a procesar.
- Manejo correcto de información o atributos del monedero.

Garantizando así la integridad de la información, la correcta sincronización entre sistemas y un manejo más robusto de escenarios offline y de falla, reduciendo la necesidad de intervenciones manuales (data fixes) y fortaleciendo la confiabilidad del sistema.