



# ***FlexPoint Integration API for Canada***

**Pay-at-the-Table**

Version: V0108+

*Programmers Guide*

Document Version: **1.2**

August 20, 2015

Document Owners:

Richard Staley  
Product Manager

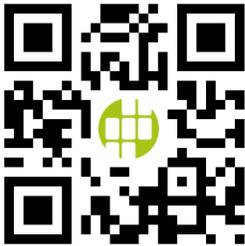


## Pivotal Product Management Press

Copyright 2015 Pivotal Payments. All rights reserved.

*Information in this document is subject to change without notice. The software described in this document is furnished under a license agreement or nondisclosure agreement. The software may be used or copied only in accordance with the terms of those agreements. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or any means electronic or mechanical, including photocopying and recording for any purpose other than the purchaser's personal use without the written permission of Pivotal Payments.*

**Pivotal Payments**  
1100 Rene-Levesque Blvd., West, Suite 9000  
Montreal QC H3B 4N4  
866-687-3722  
[www.pivotalpayments.ca](http://www.pivotalpayments.ca)



## *Revision Information*

Doc Ver	S/W Ver	Name	Title	Date of Update	Summary of Changes
1.0	0.2	Jeremy Seitz	Product Manager	November 27, 2013	Document Creation
1.1	0108	Richard Staley	Product Manager	March 16, 2015	Standardization of document format
1.2	0108	Richard Staley	Product Manager	August 20, 2015	- Added Appendix A: AID listing for mapping card type to AID numbers.

## Table of Contents

Revision Information .....	iv
Table of Contents .....	v
<b>1. Introduction .....</b>	<b>1</b>
1.1 General Overview .....	1
<b>2. Definitions and Terms .....</b>	<b>2</b>
2.1 Convention - Restaurant Checks and Tickets .....	2
<b>3. Summary of Implementation Steps .....</b>	<b>3</b>
<b>4. Messaging and Communication between FlexPoint and the POS .....</b>	<b>4</b>
4.1 Transaction Packet Format .....	4
4.1.1 Computing the Message Length .....	6
4.1.2 Calculating the LRC .....	6
4.1.2.1 Using C .....	6
4.1.2.2 Using Java .....	6
4.2 Communication Protocols between POS and FlexPoint .....	7
<b>5. Transaction Lifecycle in Full Detail .....</b>	<b>8</b>
5.1 FlexPoint Request All Tickets - (Message ID 88) Format .....	8
5.1.1 Example 1. ....	9
5.1.2 Example 2. ....	9
5.1.3 Example 3. ....	9
5.2 POS Response All Tickets - (Message ID 881) Format .....	10
5.2.1 Field 14441 Format .....	11
5.2.2 Example .....	11
5.3 FlexPoint Request Ticket Sale - (Message ID 89) Format .....	12
5.3.1 Example .....	13
5.4 POS Response Sale Transaction (Message ID 891) Format .....	13
5.4.1 POS Message ID 891 Request Record .....	13
5.4.2 Transaction ID Reference Table .....	14
5.5 FlexPoint Auth Response - (Message ID 99) Format .....	15
5.5.1 FlexPoint Auth Response Conventions .....	17
5.5.2 Example .....	17
5.6 POS Response Confirmation - (Message ID 991) Format .....	19
5.6.1 Example .....	19
5.7 Validation Rules by Message ID .....	19
<b>6. Pay-at-the-Table Important Implementation Details .....</b>	<b>21</b>
6.1 Security and Waiter Authentication .....	21
6.2 Screen Lock .....	21
6.3 Waiter Authentication .....	22
<b>7. Pay-at-the-Table Interim Summary .....</b>	<b>25</b>
7.1 Waiter Authentication and Security Recap .....	25
7.2 Check List .....	25
7.3 Cardholder Payment .....	28

- 7.3.1 Split-Payment of Guest-Checks ..... 28
- 7.4 FlexPoint Restaurant Payment Event ..... 29
  - 7.4.1 Split Guest Checks and Tips ..... 29
- 7.5 FlexPoint and POS Integration Notes ..... 29
- 8. Additional Functionality ..... 31**
  - 8.1 Managerial and Administrative Transactions ..... 31
    - 8.1.1 Hints for Implementing Managerial and Administrative Transactions ..... 32
- 9. FlexPointPOR (Power on Reset) Procedure ..... 33**
- 10. Configuration Variables ..... 34**
- Appendix A - AID's ..... 36**

## 1. Introduction

FlexPoint is Pivotal's integrated payments solution which includes support for contactless payments, EMV and point-to-point encryption\* (future enhancement). This document details the integration between a POS and the FlexPoint platform. The primary purpose of FlexPoint is to authorize payments with a host and return the response back to the POS.

The message specifications and tables in this document define the full scope and extent of integration between the POS and the FlexPoint platform.

### 1.1 General Overview

FlexPoint Pay-at-the-table functionality serves to facilitate a guest's payment of a restaurant check directly at the table. FlexPoint makes this possible by pulling and pushing information about the check and its payment to and from the restaurant POS system. It is designed for operation by wait staff on a FlexPoint that has been configured to restaurant mode. It pulls essential information from the restaurant POS system needed to charge the guest. The FlexPoint is brought to the table and the payment transaction processed.

## 2. Definitions and Terms

Terms	Definitions
<b>Ticket</b>	Tickets, Guest Checks, Invoices are all terms used to refer to the "Bill" that a restaurant would provide to a Customer for payment. For consistency FlexPoint will refer to this as a "Ticket" or "Check" throughout the document. FlexPoint does not recognize hierarchies of payables such as a specific seat on a ticket. Many POS systems will allow for users to split the Tickets by Seat so that a table with 3 people (3 seats) would end up with 3 tickets. Each of these tickets would have a unique ticket # so would be recognized just like any other ticket on FlexPoint.
<b>Retail/QSR</b>	Retail/Quick Serve Restaurant. In the context of integration, a Retail/Quick Serve Restaurant environment is one where the integration between the POS and FlexPoint is such that the payment transaction is initiated by the POS and then pushed to FlexPoint. The relationship between POS and FlexPoint is always one-to- one.
<b>POS &amp; ECR</b>	Point-of-Sale and Electronic Cash Register. 'POS' and 'ECR' are used interchangeably and may be regarded as synonyms in this document.
<b>Sales Receipt</b>	A receipt printed by the POS system. This receipt itemizes the goods or services actually purchased. Very little (if any) payment information is printed on the sales receipt—sometimes it will indicate the tender type of the payment transaction.
<b>Payment Receipt</b>	Also referred to as 'Payment Transaction Receipt' is the receipt generated by the FlexPoint.. It contains no information about the goods and services purchased (other than the total price paid), but it does contain data required by the credit card associations. Whether printed by FlexPoint or by a POS, the payment transaction receipt is created from data output by FlexPoint.

Figure 2-1 Definitions and Terms Listing

### 2.1 Convention - Restaurant Checks and Tickets

A variety of solutions exist across different POS systems to handle the splitting and sharing of guest checks. FlexPoint regards a single ticket as having a single balance owing. As stated in its definition above, ticket is the fundamental payable and the only object recognized for payment by FlexPoint.

### **3. Summary of Implementation Steps**

FlexPoint Pay-at-the-table is designed to interact with cardholders and wait staff. It first pulls a list of open tickets from the POS system and allows the waiter to choose a ticket to be paid. The waiter then confirms the amount to be paid and this amount is transmitted to the POS system. The POS then sends a sale request to FlexPoint, and FlexPoint communicates with the credit or debit networks for an authorization. FlexPoint then transmits the authorization response, Approved/Declined back to the POS and waits for confirmation from the POS that the authorization response was received.

CONFIDENTIAL

## 4. Messaging and Communication between FlexPoint and the POS

This section presents the message formats used in the exchange of information and instructions between FlexPoint and the POS. The message format will be laid out first, followed by a discussion of the message types. Sample transaction packets will also be presented at the end of this section.

### 4.1 Transaction Packet Format

The POS must send transaction packets to the interface in the format described in this section. The FlexPoint application will return responses to the POS through the interface in identical format. The Field ID and Record Data should be sent as strings. POS systems and FlexPoint must both send and receive messages that conform to the following format:

- <STX> (0x02)
- Field Separator <FS> (0x1C)
- Field ID
- Record Separator <RS> (0x1E)
- Record
- Field Separator <FS> (0x1C)
- Field ID
- Record Separator <RS> (0x1E)
- Record
- . . .
- <ETX> (0x03)
- <LRC> - 1 byte longitudinal redundancy check digit

For example, consider the following sale transaction message:  
 <STX><FS>11111<RS>00031<FS>12221<RS>89<FS>10002<RS>1<FS>10007<  
 RS>14718<ETX><LRC>

This is the raw data sent to the POS.

Array Position	Decimal Value	Character	Description
[0]	2	STX	Start Sentinel
[1]	28	FS	Field Separator
[2]	49	'1'	Field ID for MESSAGE_LENGTH
[3]	49	'1'	Field ID for MESSAGE_LENGTH
[4]	49	'1'	Field ID for MESSAGE_LENGTH
[5]	49	'1'	Field ID for MESSAGE_LENGTH

Array Position	Decimal Value	Character	Description
[6]	49	'1'	Field ID for MESSAGE_LENGTH
[7]	30	RS	Record Separator
[8]	48	'0'	Message Length
[9]	48	'0'	Message Length
[10]	48	'0'	Message Length
[11]	51	'3'	Message Length
[12]	48	'1'	Message Length
[13]	28	FS	Field Separator
[14]	49	'1'	Field ID for MESSAGE_TYPE
[15]	50	'2'	Field ID for MESSAGE_TYPE
[16]	50	'2'	Field ID for MESSAGE_TYPE
[17]	50	'2'	Field ID for MESSAGE_TYPE
[18]	49	'1'	Field ID for MESSAGE_TYPE
[19]	30	RS	Record Separator
[20]	56	'8'	Message ID for request ticket sale
[21]	57	'9'	Message ID for request ticket sale
[22]	28	FS	Field separator
[23]	49	'1'	Field ID for TRANSACTION_ID
[24]	48	'0'	Field ID for TRANSACTION_ID
[25]	48	'0'	Field ID for TRANSACTION_ID
[26]	48	'0'	Field ID for TRANSACTION_ID
[27]	50	'2'	Field ID for TRANSACTION_ID
[28]	30	RS	Record Separator
[29]	49	'1'	Transaction ID for purchase
[30]	28	FS	Field Separator
[31]	49	'1'	Field ID for TRANSACTION_AMOUNT
[32]	48	'0'	Field ID for TRANSACTION_AMOUNT
[33]	48	'0'	Field ID for TRANSACTION_AMOUNT
[34]	48	'0'	Field ID for TRANSACTION_AMOUNT
[35]	55	'7'	Field ID for TRANSACTION_AMOUNT
[36]	30	RS	Record Separator
[37]	49	'1'	Transactin amount (147.18)
[38]	52	'4'	Transactin amount (147.18)
[39]	55	'7'	Transactin amount (147.18)
[40]	49	'1'	Transactin amount (147.18)
[41]	56	'8'	Transactin amount (147.18)
[42]	3	ETX	End sentinel
[43]			LRC (not calculated in this example)

Figure 4-1 Raw Data Sent to POS

## 4.1.1 Computing the Message Length

For all messages exchanged between FlexPoint and a POS, MESSAGE\_LENGTH is a required field. This section will unpack how the message length should be counted. Consider the following sale transaction above.

The following rules should apply to the calculation of the message length:

- Message length should be expressed in a fixed block of 5 bytes. A 31 byte message should be expressed as '00031' as in the example above.
- The computation of message length should begin immediately after the expression of message count in bytes. In the case of the example above, begin counting message length after <STX><FS>11111<RS>00031. <FS> is therefore the first character to count in the sample string above.
- <FS> and <RS> are 1 byte each as per ASCII standard.
- <FS> and <RS> must be counted in the message length unless they occur before the 5 byte message length.
- <STX> is never counted because it always occurs before the message length.
- <ETX> and <LRC> are 1 bytes each as per ASCII standard.
- <ETX> and <LRC> must be counted in the message length.

## 4.1.2 Calculating the LRC

### 4.1.2.1 Using C

```
unsigned char CalculateLRC( unsigned char *pBuffer, int len )
{
    int iItem;
    unsigned char LRC = 0;
    for( iItem = 0; iItem < len; iItem++)
        LRC ^= pBuffer[ iItem ];
    return( LRC );
}
```

### 4.1.2.2 Using Java

```
private char calculateLRC(String xml)
{
    char[] buffer = xml.toCharArray();
    char lrc = 0;
    for (int ii=0; ii< buffer.length; ii++)
        lrc ^= buffer[ii];

    return lrc;
}
```

## 4.2 Communication Protocols between POS and FlexPoint

- All messages will be framed beginning with an <STX> and MESSAGE\_LENGTH, ending with an <ETX>, followed by an LRC.
- The application that receives a framed packet must respond with either an <ACK> or a <NAK> depending on whether the packet is properly formatted and has a valid LRC.
- If FlexPoint sends a response packet to the POS and receives a <NAK> then the FlexPoint will resend the response packet.
- If FlexPoint receives a <NAK> for each of 3 attempted transmissions of the same response packet then it will discard the response packet and return to its idle state.  
WARNING: If the response packet represents a successful FlexPoint transaction then FlexPoint will become out of balance relative to the POS.
- FlexPoint only supports the handling of one active transaction at time. If the POS application sends a transaction message to FlexPoint and then sends another transaction message prior to receiving a response message then FlexPoint will respond to the POS with an Error 104.
- The POS can send a reset interface transaction message in order to force the FlexPoint to reset and start receiving new transactions from the POS. Any FlexPoint pending response will be dropped and the terminal will wait for the next POS request.  
WARNING: If this reset command is sent while FlexPoint is still processing the previous transaction then the POS and FlexPoint may fall out of sync.
- FlexPoint will save a copy of the response message prior to sending it to the POS. If FlexPoint encounters a problem (i.e. FlexPoint loses power) prior to completing a successful transmission of the response message, then upon restart it will retrieve the saved copy of the packet and continue attempting to transmit it to the POS. The interface will delete the saved copy of the response packet when Message ID 991 is received from the POS.

## 5. Transaction Lifecycle in Full Detail

The functional requirements for FlexPoint operating in a restaurant environment with FlexPoint Pay-at-the-table service are fairly straightforward. The environment calls for 6 kinds of message:

Message Name	Msg ID	Description
FlexPoint Request All Tickets	88	FlexPoint's request to the POS for a list of all open tickets belonging to the operator specified in the request
POS Response All Tickets	881	POS response to FlexPoint's request for all open tickets
FlexPoint Request Ticket Sale	89	FlexPoint's request to the POS to reply with a sale transaction request for a specified amount
POS Response Sale Transaction	891	POS response to FlexPoint's sale transaction request
FlexPoint Auth Response	99	FlexPoint's response containing indication of successful approval (or decline) of the sale transaction
POS Response Confirmation	991	POS response to FlexPoint confirming successful receipt of Message ID 99

Figure 5-1 Ordered Steps of the Transaction Lifecycle

### 5.1 FlexPoint Request All Tickets - (Message ID 88) Format

This message instructs the POS system to return all open tickets. This instruction may be qualified or unqualified. The qualifying element is OPERATOR\_ID and/or TRACK\_DATA. If either is present in the message then all open tickets in the POS system associated to the ID or Track Data should be returned to FlexPoint by the POS system. If no OPERATOR\_ID or TRACK\_DATA are present in the Message ID 88 message (i.e. it is unqualified), then the POS should return all open tickets in its system. Section 6.2 will explain how a manager can configure FlexPoint in such a way that requires a waiter to authenticate by swiping an ID card or by entering a user ID.

Field ID	Field ID Code	Description	Data Type	Len	Req
Start Sentinel	<STX>	Mandatory Start Sentinel			Y
MESSAGE_LENGTH	11111	Message length in bytes	N	5	Y
MESSAGE_TYPE	12221	Message Type (Always 88)	N	3	Y
APP_NAME	10079	Name and/or version of FlexPoint software	AN	30	N

Field ID	Field ID Code	Description	Data Type	Len	Req
OPERATOR_ID	10010	Waiter/Server ID recognized by the merchant's POS system	AN	12	N
TRACK_ONE	11711	Contents of Track 1 on a mag-stripe ID card used by an employee to authenticate on the POS	AN	79	N
TRACK_TWO	11722	Contents of Track 2 on a mag-stripe ID card used by an employee to authenticate on the POS	AN	40	N
TRACK_THREE	11733	Contents of Track 3 on a mag-stripe ID card used by an employee to authenticate on the POS	AN	107	N
TICKET_NUMBER	10014	Ticket/Check Number	AN	12	N
End Sentinel	<ETX>	Mandatory End Sentinel			Y
LRC	<LRC>	Longitudinal Redundancy Check			Y

Figure 5-2 Transaction Request Message sent from the POS to the FlexPoint Interface

## 5.1.1 Example 1.

This request for all tickets using Message ID 88 passes track 1 and track 2 data from a swiped employee card (note that the message length may not be accurate):

```
<STX><FS>11111<RS>00163<FS>12221<RS>88<FS>10079<RS>ZIPTALK1.4<
FS>11711<RS>12345678901234567892123456789312345678941234567895
123456789612345678
97123456789<FS>11722<RS>12345678901234567892123456789312345678
94<ETX><LRC>
```

## 5.1.2 Example 2.

This request sends an OPERATOR ID in field 10010. In this example no mag stripe was swiped.

```
<STX><FS>11111<RS>00030<FS>12221<RS>88<FS>10079<RS>ZIPTALK1.4<
FS>10010<RS>12345678<ETX><LRC>
```

## 5.1.3 Example 3.

This request asks for every open ticket in the POS system.

```
<STX><FS>11111<RS>00030<FS>12221<RS>88<FS>10079<RS>ZIPTALK1.4<
ETX><LRC>
```

## 5.2 POS Response All Tickets - (Message ID 881) Format

This message represents the POS system’s reply to a ‘Request All Tickets’ (Message ID 88) message. It will contain a qualified or unqualified list of open tickets in the POS system.

In this message, the POS has full control over the contents presented on the FlexPoint screen by means of the SCREEN\_HEADER (field 14141), COLUMN\_HEADER (field 14142), and TICKET\_INFO fields (field 14441).

Field ID	Field ID Code	Description	Len	Req
Start Sentinel	<STX>	Mandatory Start Sentinel		Y
MESSAGE_LENGTH	11111	Message length in bytes	5	Y
MESSAGE_TYPE	12221	Message Type (Always 881)	≤ 3	Y
APP_NAME	10079	Name and/or version of POS software	≤ 12	
SCREEN_HEADER	14141	Header to display in FlexPoint’s navigation bar. Ideally would be waiter name or waiter ID number. Examples: ‘Waiter Name - Open Checks’ or ‘JESSICA TICKETS’, or ‘123456 TABLES’, etc. . .	≤ 30	
COLUMN_HEADER	14142	Optional column headers (3) for the list of open tickets. Must take the format (TICKET & TABLE & AMOUNT) where ampersands (&) separate the elements in this field. Note that the ‘Ticket’ and ‘Table’ column headers may be changed to ‘Check’, ‘Folio’, ‘Table’ or any other label desired. Ticket and Amount are mandatory fields, Table is optional. Ticket will be left-justified on the FlexPoint screen, Table centered, and Amount right-justified. Refer to section 5.2.2 for a code example, and section 7.2 for a pictorial mockup	≤ 30	
TICKET_INFO	14441	Use one of these fields for each record. Distinct open tickets should be sent to FlexPoint in distinct field 14441 records (ex: 3 open tickets should be sent in 3 field 14441 records). The format of this field is shown in section 5.2.1 below.	≤ 34	Y
ERROR_CODE	16666	The valid error codes are defined in section 9. For example return code ‘109’ if operator ID is not recognized by the POS or code ‘113’ if the waiter has no open tickets	3	N

Field ID	Field ID Code	Description	Len	Req
End Sentinel	<ETX>	Mandatory End Sentinel		Y
LRC	<LRC>	Longitudinal Redundancy Check		Y

Figure 5-3 POS Response All Tickets - (Message ID 881) Format

## 5.2.1 Field 14441 Format

Field 14441 makes it possible for the POS system to display ticket information on the FlexPoint screen. In the FlexPoint UI, the waiter is be able to scroll through the list of tickets and select one to interact with. The format of Field 14441 is defined below.

Metadata: Ticket #	&	Metadata: \$ Amount	&	Display: Ticket #	&	Display: Table #	&	Display: \$ Amount
Full ticket number	&	Amount (without symbols)	&	Ticket number recognized by waiter	&	Table #	&	Amount (with symbols)
Mandatory	M	Mandatory	M	Mandatory	M	Optional	M	Mandatory
≤ 30 (AN)	1	≤ 9 (AN)	1	≤ 8 (AN)	1	≤ 4 (AN)	1	≤ 11 (N)
N/A		N/A		Left Justified		Centered		Right Justified

Figure 5-4 Description of Field 14441 Format

**Metadata** is data needed by FlexPoint to run background functions and processes. Metadata dollar amounts should be transmitted without decimals or any other symbols. For example, the amount '\$147.18' should be transmitted as '14718'. Metadata for the Metadata Ticket # field should contain **full or unabridged** ticket numbers exactly as recognized by the POS system.

**Display Data** is the data that will be displayed on the FlexPoint screen. The waiter is expected to interpret and interact with the data presented on this screen. For this reason, strategic layout and design of the information going into these fields is important. The integrator must incorporate nuanced understanding of the restaurant and its POS system to maximize the usage of these fields for clarity and ease of operation by the waiter.

Note that all four ampersands of field 14441 are mandatory even if they separate by blank data.

## 5.2.2 Example

FlexPoint's Message ID 88 Request:

```
<STX><FS>11111<RS>00045<FS>12221<RS>88<FS>10079<RS>ZIPTALK1.4<ETX>10010<RS>12345678<ETX><LRC>
```

The following example denotes a Message ID 881 response from the POS with five open tickets belonging to operator ID 12345678:

```
<STX><FS>11111<RS>00162<FS>12221<RS>881<FS>10079<RS>YOUR POS
2.0<FS>14141<RS>12345678 OPEN
CHECKS<FS>14142<RS>CHECK&TABLE&AMOUNT<FS>14441<RS>777777777776
&17874&777776&32&$178.74<FS>14441<RS>777777777772&9995&777772&
30&$99.95<FS>14441<RS>777777777768&3776&29&$37.76<FS>14441<RS>
777777777766&417&777766&30&$4.17<FS>14441<RS>777777777761&1385
0&777761&33&$138.50<ETX><LRC>
```

For clarity, the exact same message is laid out below in a format that is easier read:

```
<STX>
<FS>11111<RS>00162
<FS>12221<RS>881
<FS>10079<RS>YOUR POS 2.0
<FS>14141<RS>12345678 OPEN TABLES
<FS>14142<RS>CHECK&TABLE&AMOUNT
<FS>14441<RS>777777777776&17874&777776&32&$178.74
<FS>14441<RS>777777777772&9995&777772&30&$99.95
<FS>14441<RS>777777777768&3776&29&$37.76
<FS>14441<RS>777777777766&417&777766&30&$4.17
<FS>14441<RS>777777777761&13850&777761&33&$138.50
<ETX><LRC>
```

### 5.3 FlexPoint Request Ticket Sale - (Message ID 89) Format

This message instructs the POS system to respond with a Sale Transaction Request to FlexPoint for a specific ticket and a specific dollar amount.

Field ID	Field ID Code	Description	Data Type	Len	Req
Start Sentinel	<STX>	Mandatory Start Sentinel			Y
MESSAGE_LENGTH	11111	Message length in bytes	N		Y
MESSAGE_TYPE	12221	Message Type (Always 89)	N	≤ 3	Y
EXTERNAL_APP_NAME	10079	Name and/or version of FlexPoint software	AN	≤ 30	N
TRANSACTION_AMOUNT	10007	Transaction amount (\$\$\$\$\$çç)	N	≤ 9	Y
OPERATOR_ID	10010	Waiter/Server ID recognized by the merchant's POS system	AN	≤ 12	N

Field ID	Field ID Code	Description	Data Type	Len	Req
TRACK_ONE	11711	Contents of Track 1 on a mag-stripe ID card used by an employee to authenticate on the POS	AN	≤ 79	N
TRACK_TWO	11722	Contents of Track 2 on a mag-stripe ID card used by an employee to authenticate on the POS	AN	≤ 40	N
TRACK_THREE	11733	Contents of Track 3 on a mag-stripe ID card used by an employee to authenticate on the POS	AN	≤ 107	N
TICKET_NUMBER	10014	Ticket/Check Number	AN	≤ 12	N
End Sentinel	<ETX>	Mandatory End Sentinel			Y
LRC	<LRC>	Longitudinal Redundancy Check			Y

Figure 5-5 Transaction Request Message sent from the POS to FlexPoint

### 5.3.1 Example

This request from FlexPoint to the POS pulls a single ticket from the POS. In this example, OPERATOR\_ID has been keyed in by the waiter, and field 10014 contains the request for the specific ticket.

```
<STX><FS>11111<RS>00064<FS>12221<RS>89<FS>10079<RS>ZIPTALK1.4<FS>10010<RS>123456789012<FS>10014<RS>12345678<ETX><LRC>
```

## 5.4 POS Response Sale Transaction (Message ID 891) Format

This message instructs FlexPoint to initiate a sale/purchase transaction (Transaction ID 1 in Field 10002) for the amount of TRANSACTION\_AMOUNT (Field 10007) in reference to TICKET\_NUMBER (Field 10014)

### 5.4.1 POS Message ID 891 Request Record

Field ID	Field ID Code	Description	Data Type	Len	Req
Start Sentinel	<STX>	Mandatory Start Sentinel			Y
MESSAGE_LENGTH	11111	Message length in bytes	N		Y
MESSAGE_TYPE	12221	Message Type (Always 89)	N	≤ 3	Y
EXTERNAL_APP_NAME	10079	Name and/or version of FlexPoint software	AN	≤ 30	N

Field ID	Field ID Code	Description	Data Type	Len	Req
TRANSACTION_ID	10002	Transaction Type. See table of Transaction IDs in the next section	N	≤ 3	Y
TRANSACTION_AMOUNT	10007	Transaction amount (\$\$\$\$\$cç)	N	≤ 9	Y
OPERATOR_ID	10010	Waiter/Server ID recognized by the merchant's POS system	AN	≤ 12	N
MER_REF_NUMBER	10012	Reference number from the POS. Information placed in this field will appear in transaction detail reporting available in Pivotal360.	AN	26	N
TICKET_NUMBER	10014	Ticket/Check Number	AN	≤ 26	Y
POS_TRANSACTION_DATE	10027	YYYYMMDD	N	8	N
POS_TRANSACTION_TIME	10028	HHMMSS (24hr clock)	N	6	N
INVOICE	90001	Invoice number received from FlexPoint. Used for Voids, Refunds, and for Pre-Auth Completions ONLY to refer to the initial transaction	AN	7	N
End Sentinel	<ETX>	Mandatory End Sentinel			Y
LRC	<LRC>	Longitudinal Redundancy Check			Y

Figure 5-6 Transaction Request Message sent from POS to FlexPoint

## 5.4.2 Transaction ID Reference Table

The following transaction types are recognized by FlexPoint. A single 'Trans ID' must be populated in field 10002 of the POS Response Sale Transaction (Message ID 891).

Transaction Name	Trans ID	Card Type(s)	Description
Purchase	1	DB, CR	To affect a sale to the cardholder
Refund	16	DB, CR	To credit the cardholder with funds
Pre-Auth	2	CR	To hold funds for the amount specified. It is the POS system's responsibility to add 'over lift' to the amount when needed. Pre-Auths may be voided by sending a Pre-Auth Completion for \$0. Pre-Auths not completed within the allotted timeframe (typically 5 days) will

Transaction Name	Trans ID	Card Type(s)	Description
			expire
Pre-Auth Completion	3	CR	Used to capture the final amount of a credit-card Pre-Authorization
Void	10, 21, 26	DB, CR	(10) Used for cancelling a credit card purchase, refund, pre-auth completion, or force transaction that has not been settled (i.e. that is in the current batch) (21) Debit Purchase Correction (26) Debit Refund Correction
Cancel Transaction	0	DB, CR	Prompts FlexPoint to cancel the current transaction. This transaction request will be denied if FlexPoint already received a response from the host
Close Batch	41	DB, CR	To request settlement at the host for transactions in the current batch
Reset Interface	66	DB, CR	To reset a frozen or unresponsive connection to FlexPoint. See section 10 for more information

Figure 5-7 Transaction ID

## 5.5 FlexPoint Auth Response - (Message ID 99) Format

This message is created by FlexPoint *after* it has contacted the host for authorization. This final message in the transactional flow contains the response from the authorizer. An authorization response will be provided for every Message ID 891 message sent by the POS, unless there is an error, in which case an error code will be returned in fields 11010 and 21010.

Field ID	Field ID Code	Description	Len	DB, CR
Start Sentinel	<STX>	Start Sentinel		
MESSAGE_LENGTH	11111	Message length in bytes		DB, CR
MESSAGE_TYPE	12221	Message Type (Always 99)	≤ 3	
MER_NAME	11800	DBA Name	≤ 24	DB, CR
MER_STREET_ADDR	11801	Merchant Street Address	≤ 24	DB, CR
MER_CITY_PROV_PC	11802	Merchant City, Province, Postal Code	≤ 24	DB, CR
MER_TEL	11803	Merchant Telephone Number	≤ 20	DB, CR
FIXED_MESSAGE_1	11804	"TRANSACTION RECORD"	≤ 24	DB, CR
TRANS_TYPE	11805	(Ex. 'Sale', 'Void', 'Refund', etc. . . )	≤ 24	DB, CR
CARD_NUMBER	11001	Card Account Number (masked with last 4 digits displayed)	≤ 19	DB, CR
TENDER_TYPE	11006	Credit (0), Debit (1), Gift/Loyalty (2)	≤ 24	DB, CR

Field ID	Field ID Code	Description	Len	DB, CR
CARDHOLDER_DEBIT_ACCOUNT	11806	Ex. Chequing, Savings, Other	≤ 24	DB
ENTRY_METHOD	11807	'Keyed', 'Swiped', 'Dipped', or 'Contactless'	≤ 24	DB, CR
TIP_AMOUNT	11808	Tip Amount	≤ 9	DB, CR
SURCHARGE_AMOUNT	11809	Transaction Surcharge Amount	≤ 9	DB
TRANSACTION_AMOUNT	11811	Transaction Amount excluding surcharge and tips added by cardholder on FlexPoint (value sent from POS in field 10007)	≤ 9	DB, CR
TOTAL_AMOUNT	11003	Total amount charged to cardholder	≤ 7	DB, CR
INVOICE_NUMBER	11009	Invoice Number Generated by FlexPoint or by Host	≤ 7	DB, CR
TID	11812	Terminal ID Number	≤ 8	DB, CR
OPERATOR_ID	11813	Clerk/Server/Driver ID Number (field 10010 from POS outgoing request)	≤ 12	DB, CR
MER_REF_NUMBER	11814	Merchant's own Reference Number (field 10012 from POS outgoing request)	≤ 24	DB, CR
SEQUENCE	11815	Host Sequence Number	≤ 7	DB, CR
AID <sup>1</sup>	11816	Application ID	≤ 24	DB, CR
CRYPTOGRAM_TYPE_INFO	11817	Cryptogram Type and Info	≤ 24	DB, CR
TSI	11818	Transaction Status Information	≤ 24	DB, CR
HOST_TRANS_DATE	11007	Transaction Date in YYYY-MM-DD Format	10	DB, CR
HOST_TRANS_TIME	11008	Transaction Time in HH:MM:SS (24-hour) Format	8	DB, CR
AUTHORIZATION_NUMBER	11004	Authorization/Approval Code	≤ 6	DB, CR
ISO_RESPONSE_CODE	11819	2 Digit ISO Code. Will be present for Debit only. Should not be used to determine Transaction Approved/Declined disposition	2	DB, CR
POS_RESULT_CODE	11011	3 Digit Host Response Code. Used to determine transaction approved/declined disposition. '000' indicates an APPROVED transaction, all other values indicate a NOT APPROVED	3	DB, CR

Field ID	Field ID Code	Description	Len	DB, CR
		transaction		
FP_COMM_ERROR	11010	For Communication Errors, Message Formatting Errors, or User-Initiated Cancellations	3	DB, CR
FP_COMM_ERROR_DISPLAY	21010	Communication Error message to display on FlexPoint screen.	≤ 30	DB, CR
POR_RESPONSE	11888	Response provided only when a 'Reset Interface' (TRANSACTION ID 66) is initiated by the POS.	≤ 99	DB, CR
END SENTINEL	<ETX>	End Sentinel		
LRC	<LRC>	Longitudinal Redundancy Check		

Figure 5-8 FlexPoint Auth Response Format (Message ID 99)



**Note: 1. For a list of AID's please see Appendix A.**

## 5.5.1 FlexPoint Auth Response Conventions

- FlexPoint will return a response message (Message ID 99) or error code for each request submitted by the POS (Message ID 891).
- The DB, CR column in the table above indicates whether the specific field will be returned in the FlexPoint response message for Debit transactions only (DB) or for both Debit transactions and Credit transactions (CR, DB), or neither in select cases such as POR Resets.
- Parse POS\_RESULT\_CODE (field 11011) to know if a transaction is approved or declined. '000' means approved, all other codes mean not approved.

## 5.5.2 Example

This sample transaction packet is for a \$35.52 sale and a \$6.00 tip. The transaction was approved and the tender type was debit:

```
<STX><FS>11111<RS>XXXXX<FS>10079<RS>YOUR POS
2.0<FS>12221<RS>99<FS>11800<RS>People'sRestaurant<FS>11801<RS>
123 General Street<FS>11802<RS>Moose Jaw, SK,
S7K1N6<FS>11803<RS>306-752-
2175<FS>11805<RS>Purchase<FS>11001<RS>XXXXXXXXXXXX5666<FS>1100
6<RS>1<FS>11806<RS>CHEQUING<FS>11807<RS>DIPPED<FS>11808<RS>600
<FS>11809<RS><FS>11811<RS>3552<FS>11003<RS>4152<FS>11009<RS>12
34567<FS>11812<RS>12345678<FS>11813<RS>123456789012<FS>11814<R
S>432109876543210987654321<FS>11815<RS>0000001<FS>11816<RS>A00
00002771010<FS>11817<RS>80 80 00 80
```

```
00<FS>11818<RS>6800<FS>11007<RS>2015-01-  
01<FS>11008<RS>23:59:59<FS>11004<RS>654321<FS>11819<RS>00<FS>1  
1011<RS>000<FS>11010<RS><FS>21010<RS><FS>11888<RS><ETC><LRC>
```

Laid out in more reader-friendly format:

```
<STX>  
<FS>11111<RS>XXXXX  
<FS>10079<RS>YOUR POS 2.0  
<FS>12221<RS>99  
<FS>11800<RS>People's Restaurant  
<FS>11801<RS>123 General Street  
<FS>11802<RS>Moose Jaw, SK, S7K 1N6  
<FS>11803<RS>306-752-2175  
<FS>11805<RS>Purchase  
<FS>11001<RS>XXXXXXXXXXXXX5666  
<FS>11006<RS>1  
<FS>11806<RS>CHEQUING  
<FS>11807<RS>DIPPED  
<FS>11808<RS>600  
<FS>11809<RS>  
<FS>11811<RS>3552  
<FS>11003<RS>4152  
<FS>11009<RS>1234567  
<FS>11812<RS>12345678  
<FS>11813<RS>123456789012  
<FS>11814<RS>432109876543210987654321  
<FS>11815<RS>0000001  
<FS>11816<RS>A0000002771010  
<FS>11817<RS>80 80 00 80 00  
<FS>11818<RS>6800  
<FS>11007<RS>2015-01-01  
<FS>11008<RS>23:59:59  
<FS>11004<RS>654321  
<FS>11819<RS>00  
<FS>11011<RS>000  
<FS>11824<RS>00 APPROVED 000 THANKYOU  
<FS>11010<RS>  
<FS>21010<RS>  
<FS>11888<RS>  
<ETC>  
<LRC>
```

## 5.6 POS Response Confirmation - (Message ID 991) Format

The POS system sends this message to FlexPoint to confirm successful receipt of the FlexPoint Auth Response (Message ID 99). Note: FlexPoint will continue to send Message ID 99 to the POS every 3 seconds until it successfully receives this Message ID 991 from the POS.

Field ID	Field ID Code	Description	Data Type	Len	Req
Start Sentinel	<STX>	Mandatory Start Sentinel			Y
MESSAGE_LENGTH	11111	Message length in bytes	N		Y
MESSAGE_TYPE	12221	Message Type (Always 89)	N	≤ 3	Y
EXTERNAL_APP_NAME	10079	Name and/or version of FlexPoint software	AN		N
INVOICE	10007	Transaction amount (\$\$\$\$\$c)	AN	7	N
End Sentinel	<ETX>	Mandatory End Sentinel			Y
LRC	<LRC>	Longitudinal Redundancy Check			Y

Figure 5-9 POS Response Confirmation - Message ID 991 Format

### 5.6.1 Example

```
<STX><FS>11111<RS>00047<FS>12221<RS>991<FS>10079<RS>YOUR POS  
2.0<FS>90011<RS>1234567<ETX><LRC>
```

## 5.7 Validation Rules by Message ID

The following table itemizes the mandatory fields of the 6 different Message ID types. All messages must include the Start Sentinel, End Sentinel, and LRC – these constant and static elements of a message are not listed among the mandatory fields in the table below. This table indicates what to expect to receive from FlexPoint in messages from FlexPoint to the POS, and also to understand what must be sent to FlexPoint by the POS.

Trans Name	Msg ID	Mandatory Fields	Integration Note
FlexPoint Request All Tickets	88	11111, 12221, *10010, *11711, *11722, *11733	*Note that all open tickets in the POS system should be returned in the POS response to this request unless qualified by an OPERATOR_ID in field 10010 or by track data from an employee magcard in any or all of fields (11711, 11722, 11733)
POS Response All Tickets	881	11111, 12221, *14441	*At least one field 14441 is required corresponding to at least one open ticket in the POS system unless an ERROR_CODE is present in field

Trans Name	Msg ID	Mandatory Fields	Integration Note
			16666
FlexPointRequest Ticket Sale	89	11111, 12221, 10007, 10014, *10010, *11711, *11722, *11733	*At least one of these fields is mandatory if an Operator ID or Employee ID should be passed between FlexPoint and the POS
POS Response Sale Transaction	891	11111, 12221, 10002, 10007, 10014, *10010, *11711, *11722, *11733	*Please refer to the table in section 5.4.2 for a complete list of transaction types.
FlexPoint Auth Response	99	11111, 12221 *See Note	*Mandatory fields are listed in section 5.5 and its subsections.
POS Response Confirmation	991	11111, 12221, 90011	

Figure 5-10 Mandatory Fields by Message Type

CONFIDENTIAL

## 6. Pay-at-the-Table Important Implementation Details

This section will address several items of importance pertaining to the bar and restaurant Pay-at-the-Table environment. Those items focus on user security and access, as well as on ticket numbering.

### 6.1 Security and Waiter Authentication

Restaurants and their POS systems have defined security policies in place. FlexPoint allows for the transmission of operator ID and/or track data in communication to the POS system. It is incumbent on the POS system to interpret the operator ID and/or track data sent from FlexPoint and perform its own waiter authentication.

When delivered to a merchant location, FlexPoint arrives with pre-configured settings that have been optimized for Integration to POS systems and for a FlexPoint Pay-at-the-table environment. The waiter authentication settings on FlexPoint may be adjusted by a user with managerial access to FlexPoint. Section 11 discusses the configuration parameters that may be adjusted by a manager on a FlexPoint device.

Whether by unique waiter login or by password shared among all restaurant staff, it is advisable to apply at least one security measure on FlexPoint in order to prevent unauthorized use. FlexPoint offers 5 options for security and waiter authentication. The first security option – Screen Lock – is unique insofar as it is applied only on the FlexPoint device and is independent of the POS. The other four security and authentication options involve the POS. All are discussed in detail in this section. Use the ‘Screen Lock’ and ‘Waiter Authentication’ configuration parameters in the FlexPoint menu system, described in section 11 to apply the desired kind of waiter security and authentication.

### 6.2 Screen Lock

Screen lock is a security access function requiring entry of a common password shared by all authorized users of FlexPoint. It must be set for each FlexPoint in use. This security feature is applied at the FlexPoint level only, and there is no authentication at the POS with this security feature. Screen Lock may be turned on or off on FlexPoint’s configuration parameters, and when it is on, the unlock screen will resemble the following:

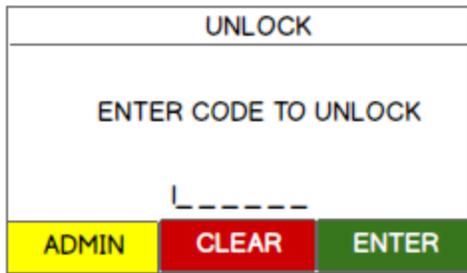
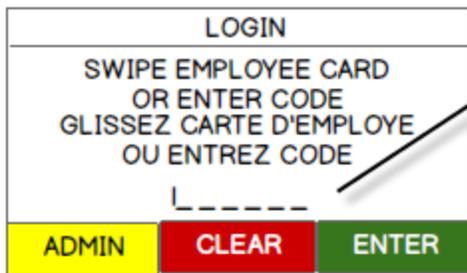


Figure 6-1 Unlock Screen

It is advisable to use screen lock in an environment where tables are shared among all wait staff and where there are no unique waiter IDs or login.

## 6.3 Waiter Authentication

The remaining four options for waiter security and authentication involve FlexPoint's interaction with the POS system and are described in detail below. The solutions consist of swiping a card containing a mag stripe or the keying-in of a waiter ID code on the FlexPoint device. As presented in section 5.1, all track data on the card, or IDs keyed in by the waiter are returned to the POS in the Message ID 88 request.



Keyed-In Waiter IDs populated in Field 10010,  
Track One Data populated in Field 11711,  
Track Two Data populated in Field 11722,  
Track Three Data populated in Field 11733,  
...of the Transaction ID 88 message.

Figure 6-2 Login Screen

It is important to reiterate that FlexPoint performs no authentication of the employee ID or mag stripe data. The POS system shall receive and validate the login info embedded in the Message ID 88 message, and if the login ID is accepted by the POS, the POS shall respond with a list of tickets. If the login ID is not accepted by the POS, it should respond with the appropriate error code (see table of error codes in section 9).

The five kinds of Waiter authentication at the POS are as follows:

Authentication Name	Description
No Authentication	If calibrated as such, the Waiter will not be prompted by FlexPoint to provide an operator ID or track data. Consequently, 'Pull All Tickets' (Message ID 88) and 'Pull Ticket' (Message ID 89) requests sent by FlexPoint to the

Authentication Name	Description
	POS will contain neither of these.
<b>Operator ID Authentication</b>	If activated, FlexPoint will require the operator to enter a numeric ID code/password of 4 - 12 digits in length. FlexPoint will not evaluate the ID entered. FlexPoint will simply pass the datum in the appropriate field (10010) of the various message formats. The POS system shall scrub the ID and decide if it is valid. If valid the POS should respond with the info requested by FlexPoint. If invalid, the POS system should reject the transaction by returning an error code in its Message ID 881 response to FlexPoint described in section 5.2.
<b>TRACK X Authentication</b>	If activated, FlexPoint will require the operator to swipe a card containing a mag stripe for authentication. FlexPoint will send any track data available on the card in their respective fields: 11711, 11722, 11733 of the Transaction 88 or 89 message. The POS system shall scrub the data and decide if it is valid. If valid, the POS should respond with the info requested by FlexPoint in the Message ID 88. If invalid, the POS system should reject the transaction by returning an error code in its Message ID 881 response to FlexPoint described in section 5.2.
<b>TRACK Authentication [or] OPERATOR ID Authentication</b>	This solution is a hybrid of the two preceding authentication methods. This setting renders it mandatory for a waiter to either swipe his ID card or key in an OPERATOR ID.
<b>TRACK X Authentication [and] OPERATOR ID Authentication</b>	This solution is also a hybrid of the two preceding authentication methods. This setting renders it mandatory for a waiter to swipe his ID card and key in an OPERATOR ID. Note that in this context, while that field is labeled 'OPERATOR ID' it can be used with equal efficacy as a password field associated to the swiped ID card.

Figure 6-3 Five Waiter Authentications

The following graphic illustrates the field location mappings on the FlexPoint screen mapped to a Message ID 881 response.

12345678 OPEN CHECKS		
CHECK	TABLE	AMOUNT
(1)...777776	32	\$178.74
(2)...777772	30	\$99.95
(3)...777768	29	\$37.76
(4)...777766	30	\$4.17

CANCEL SCROLL SELECT

Field 14141  
Field 14142  
Field 14441

Figure 6-4 Open Checks Screen

CONFIDENTIAL

## 7. Pay-at-the-Table Interim Summary

This section consists of an interim summary of the steps in the transaction lifecycle. This section is enhanced by wire-frames to aide visualization of the transaction flow, messaging, and waiter experience (UI). The following steps have thus far been covered in the Restaurant Pay-at-the-table lifecycle:

### 7.1 Waiter Authentication and Security Recap

As was presented in section 6, two types of security are offered on FlexPoint: Screen Lock and 'Waiter Authentication'. Screen Lock is to gain access to the terminal by a single password shared by all staff, and Waiter Authentication sees unique IDs (by password or mag stripe card) assigned to each employee. The ID is transmitted to the POS and the POS decides if the ID submitted is valid or invalid. 'Screen Lock' authentication only results in FlexPoint sending a Message ID 88 request for all tickets, whereas 'Waiter Authentication' results in FlexPoint sending a Message ID 88 that has been qualified with unique Operator ID or Track Data from an employee ID card. Depending on the security and authentication configuration settings, a waiter using FlexPoint would see one of the following screens:

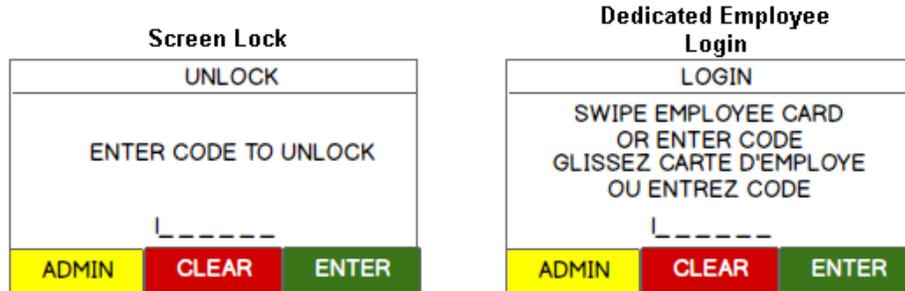


Figure 7-1 Unlock and Login Screens

### 7.2 Check List

Upon successful authentication on the POS, the POS shall send a Message ID 881 message as a response to FlexPoint's Message ID 88 request. The screens should typically resemble one of the two below.

JESSICA OPEN CHECKS		
Folio	Table	Amount
(1)6262802	tbl 27	\$125.99
(2)6262807	tbl 29	\$68.75
(3)6262811	tbl 22	\$34.50
(4)6262826	tbl 25	\$101.23

CANCEL SCROLL SELECT

OPEN CHECKS		
(1)...344564	tbl 67	\$134.15
(2)...344569	tbl 71	\$56.20
(3)...344570	tbl 72	\$56.20
(4)...344572	tbl 75	\$56.21
(5)...344599	tbl 68	\$195.67

CANCEL SCROLL SELECT

Figure 7-2 Open Checks by Employee & Open Checks Screen

Note the differences between the two wireframes where the left frame contains a screen header and column headers, while the right frame contains only a screen header. The left frame is an example of a Message ID 881 response from the POS system that has been qualified by a unique employee ID in the original FlexPoint Message ID 88. The frame on the right makes use of the screen header field ('OPEN CHECKS'), but does not use the field dedicated to column headers. The frame on the right may also be unqualified, which is to say, it is a list of all open tickets in the POS system rather than the open tickets belonging to a single waiter. . .

The waiter must now select the ticket he wishes to settle by touching the screen or by keying-in the corresponding ticket number displayed in brackets.

The following series of screens lead up to the payment event.

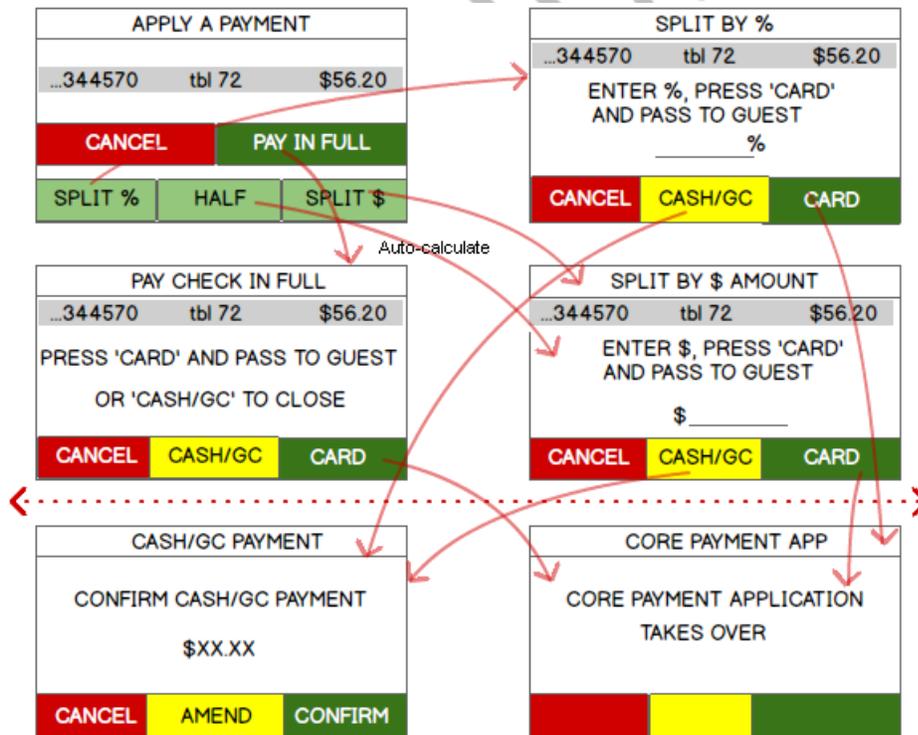


Figure 7-3 Payment Flow Screens

A few comments and observations about this flow:

- The 'APPLY A PAYMENT' screen is displayed immediately upon the waiter's selection of a specific ticket for payment.
- While less common than card payments, having the ability to close an amount to cash will make it much easier for a waiter to manage decrementing ticket balances when tickets are split into multiple payments. This will ultimately reduce waiter errors and financial loss.
- The 'CONFIRM CASH/GC PAYMENT' screen is necessary to prevent accidental keying of the wrong button(s).
- IMPORTANT. If 'CARD' is the selected payment method of choice, at the point of the dotted red line in the diagram above FlexPoint will send a Message ID 89 message to the POS to prompt it to return an outgoing transaction request (Message ID 891). Once FlexPoint receives the Message ID 891 request from the POS for a sale transaction, FlexPoint's core payment application takes over and guides the cardholder through the payment process.
- On the 'Split by %' and 'Split by \$ Amount' screens, the prompts instruct the waiter to press the 'CARD' button to initiate a payment event by credit or debit card. But it should also be obvious that despite any message on screen, that if the waiter wishes to register a sale by cash instead of by card, that he would not press the 'CARD' button, but rather press the 'CASH/GC' button, and would not pass the FlexPoint to a guest paying in cash.
- If a payment is by Credit, Debit, Gift, or Loyalty, then the server must choose the 'CARD' button. If cash or gift certificate is used, the server is to close the transaction out to 'CASH/GC'. It is expected that gift and loyalty transactions shall be card-based, and therefore FlexPoint will recognize a gift or loyalty card that has been swiped, and distinguish it from a credit or debit transaction. FlexPoint shall also populate the correct payment medium code in field 11006 - 'TENDER TYPE' of the FlexPoint Response Record §5.5.
- The server may opt to close cash transactions directly on the POS terminal instead of on FlexPoint if such a function is possible on the POS. The ability to close to cash on FlexPoint is made possible to aid the server when keeping track of large guest checks that have been split into many payments.
- As soon as the waiter presses the 'CARD' button, FlexPoint shall send a Message ID 89 message to the POS. If the waiter indicated the full amount was to be paid, then the full amount of the ticket will be sent in Message ID 89. If the ticket is split by the waiter, the partial amount keyed in by the waiter must be specified in the transaction message.
- Hitting 'CANCEL' at any point should return to the Screen Lock or to the Employee Login screen. This assumes that at least one of these two security functions is activated on FlexPoint.

## 7.3 Cardholder Payment

As mentioned in the bullet points above, once the POS sends the Message ID 891 request to charge the cardholder, the core payment application takes control of FlexPoint and interacts with the cardholder. It is here where there cardholder will have the opportunity to leave a tip and agree to surcharges (if permitted and applicable).

It is immediately after the cardholder interaction portion of the transaction where FlexPoint connects to the host to procure an authorization. The response message containing among many things, approval or decline code, tip amount, and other important information is displayed on the FlexPoint screen and returned to the POS at the same time.

There are several possible outcomes when the guest makes a payment - i.e. once the FlexPoint is handed over to the cardholder. These are the possibilities:

Permutation	Action
Customer changes mind, decides to pay Cash (or gift-certificate for example)	Close ticket out to CASH/GC on FlexPoint (optional) or directly on restaurant POS. If closed on FlexPoint, a response message shall be sent to the restaurant POS with the proper Tender Type
Customer changes mind, decides to split the check for a different amount	Cancel out of current transaction and recommence entire transaction flow beginning from waiter authentication. This scenario will not happen very often
Customer pays ticket in entirety	Response message sent from FlexPoint to restaurant POS confirming payment with approved or declined response
Customer card declines	Server starts a new transaction by calling up guest-check and requesting an alternative card for payment

Figure 7-4 Payment Flow Choices - Customer

### 7.3.1 Split-Payment of Guest-Checks

Until the cardholder payment takes place, the POS and FlexPoint do not know how the guest-check might be further subdivided. The amounts paid in a split-payment scenario will only be known with certainty at the time the guest-check is presented for payment at the table (although on the restaurant POS a server may certainly subdivide a master check into several guest checks for the subdivisions he knows or that have been explicitly requested by the guest in advance.) But since the restaurant POS system sends the total amount of a given guest-check to the FlexPoint for payment, the server needs a way to change the amount to be authorized for payment. The payment of a split guest-check therefore unfolds in the following ordered way:

1. We assume the server has authenticated to call up the list of open tickets in the procedure described in section 6.

2. Server selects ticket.
3. Server indicates whether ticket is to be paid in full or if it is to be split.
4. Server splits ticket(s) as required.
5. Server presses 'CARD' to make a card payment, unless the ticket is being closed out to cash or gift certificate.
6. Now that the correct amount for payment is established and confirmed, FlexPoint prompts the restaurant POS system to return an outgoing message request as described in section 5.4.
7. The outgoing message from the restaurant POS system to FlexPoint shall contain the following mandatory data elements: TRANSACTION\_ID, BASE\_AMOUNT, OPERATOR\_ID, TICKET\_NUMBER.
8. Upon receipt of the ticket's transaction information, FlexPoint commences the payment event and the FlexPoint is handed over to the guest for secure payment.

## 7.4 FlexPoint Restaurant Payment Event

### 7.4.1 Split Guest Checks and Tips

Suppose a guest-check of \$200 and upon presentation to the table, two cardholders decide to split the check evenly (the sharing ratio is not important to this example). Cardholder 1 pays \$100 and leaves a \$20 tip, cardholder 2 does the same thing. It is important that upon completion of payment to cardholder 1 and receiving the transaction response from the FlexPoint, the restaurant POS system must decrement the total amount owed on the check by \$100, and not by \$120. The tip amount will be passed in the response message, but in a separate field. And so while each payment may be viewed as one for \$120 by the payment networks, this transaction is actually one for \$100 + \$20 and should be distinguished as such in the response messaging and in the restaurant POS system. The following fields, found in the FlexPoint transaction response Message ID 99 described in section 5 are the response fields relevant to this discussion about TIPS:

Field ID & Field Name	Description
11811 - TRANSACTION_AMOUNT	Transaction amount excluding any tips input by cardholder and/or surcharge(s) levied on FlexPoint
11808 - TIP_AMOUNT	Tip amount input by cardholder
11003 - TOTAL_AMOUNT	Total amount charged to cardholder comprised of Trans amount, tip, surcharge, cashback

Figure 7-5 Relevant Tip Fields

## 7.5 FlexPoint and POS Integration Notes

While the FlexPoint interface was designed to be generic and simple, several technical requirements must obtain in order to successfully integrate a restaurant point-of-sale system with the Pay-at-the-Table module of the FlexPoint interface. The requirements are:

1. FlexPoint Wi-Fi devices are in use for integration.
2. The restaurant POS system has been calibrated to use the FlexPoint Transaction Request and Authorization Response messaging specification described in this document.
3. The POS system's fundamental object of payment is the ticket. 'Check' and 'Ticket' are synonyms.
4. The restaurant POS system is capable of subdividing a ticket/check into sub-checks.
5. Ticket numbers issued by the Restaurant POS system must be unique across the restaurant per batch at the very least. It is preferable to preserve uniqueness of Ticket numbers across an even longer time span if possible (ex. weekly, monthly, etc.)
6. FlexPoint should display the Operator ID, Server Name, table #, and check # on screens used by servers (especially the screen immediately following the transmission of a transaction from the restaurant POS to the FlexPoint - prior to the server passing the FlexPoint device to the guest).
7. The POS system supports pushing and pulling of data from FlexPoint.
8. **Note that Sub-checks are for bill-printing convenience only**, and are not necessarily tracked by the restaurant POS system as subdivided amounts. By managing its own tickets, the POS system remains the locus of logic used to track and manage payments. FlexPoint will process payments for any amount(s) sent by the POS system.

CONFIDENTIAL

## 8. Additional Functionality

### 8.1 Managerial and Administrative Transactions

Restaurants may operate for weeks without the need to refund or void a transaction. Nevertheless, the need occasionally arises and the FlexPoint is equipped to support a large variety of transaction types that include refunds and voids, but also batch-settlement and FlexPoint reset commands. This section will discuss this aspect of the integration.

In keeping with the theme of this software design, the POS may initiate any desired managerial or administrative transactions. The POS shall also be responsible for granting the level of user access needed to initiate these sorts of transactions. In a restaurant, waiters may not be authorized to issue refunds or close a batch, but the managers may be authorized. These user privileges should be programmed into the user hierarchy and security settings defined in the POS.

The following table is repeated from section 5.4.2. Much of the focus of this document has been on Transaction ID 1 (sale/purchase) thus far. The focus in this subsection is on the other transaction types – some that may be restricted to managers only, others that may be incorporated into automated processes on the POS such as scheduled or event-triggered batch-outs at the host, or Reset Interface. Brief suggestions for implementation will follow this table:

Transaction Name	Trans ID	Card Type(s)	Description
Purchase	1	DB, CR	To affect a sale to the cardholder
Refund	16	DB, CR	To credit the cardholder with funds
Pre-Auth	2	CR	To hold funds for the amount specified. It is the POS system's responsibility to add 'over lift' to the amount when needed. Pre-Auths may be voided by sending a Pre-Auth Completion for \$0. Pre-Auths not completed within the allotted timeframe (typically 5 days) will expire
Pre-Auth Completion	3	CR	Used to capture the final amount of a credit-card Pre-Auth
Void	10, 21, 26	DB, CR	(10) Used for cancelling a credit card purchase, refund, pre-auth completion, or force transaction that has not been settled (i.e. that is in the current batch) (21) Debit Purchase Correction (26) Debit Refund Correction
Cancel Transaction	0	DB, CR	Prompts FlexPoint to cancel the current transaction. This transaction request will be denied if FlexPoint already

Transaction Name	Trans ID	Card Type(s)	Description
			received a response from the host
Close Batch	41	DB, CR	To request settlement at the host for transactions in the current batch
Reset Interface	66	DB, CR	To reset a frozen or unresponsive connection to the FlexPoint terminal. See section 10 for more information

Figure 8-1 Transaction ID

## 8.1.1 Hints for Implementing Managerial and Administrative Transactions

- Send a Message ID 891 message to FlexPoint to initiate the transaction at the host.
- FlexPoint will respond with a Message ID 99 message or an error code.
- The POS should respond with a Message ID 991 to confirm receipt of the auth response.
- Insert the desired Transaction ID code according to the table above into field 10002 of the 891 message.
- To access these features directly on FlexPoint (instead of sending the message request from the POS), login as a managerial or admin user to gain privilege to initiate the transactions described in the table.
- Close batches (Transaction ID 41) may occur as often as desired.

CONFIDENTIAL

---

## 9. FlexPointPOR (Power on Reset) Procedure

A POS that successfully transmits a TRANSACTION\_ID of '66' to reset FlexPoint will cause the terminal to reboot.

If a transaction is submitted by the POS and no FlexPoint response or error code is returned after some requisite waiting period determined by the POS, then the POS should initiate a reset and wait for the terminal to complete a POR procedure.

Upon completion of the rebooting procedure, FlexPoint will send a POR transaction response packet to the POS. The packet is confirmation that the terminal is ready to receive transactions. It is at this point that the POS should resend the last transaction (which had failed to be processed successfully and is the reason for the POR transaction).

The POR response transaction packet is also sent by FlexPoint as an unsolicited message each time the terminal is booted (regardless of whether it is prompted to reset by the POS via TRANSACTION ID 66).

CONFIDENTIAL

## 10. Configuration Variables

The following settings may be adjusted to modify the functionality of the integration using the specific configuration parameters defined in this table. They may be adjusted in the admin settings screens on FlexPoint.

Config Label	Description	Values	Default
RETAIL_MODE	Activate or deactivate integration	0 - Off 1 - On	Off
BEEP	Enable/disable application interface beep: (a) when a packet is received from the POS (b) when a packet is sent to the POS. N.B. This parameter is intended for troubleshooting purposes and should not normally be enabled in a production environment	0 - Off 1 - On	Off
RECVTO	POS receive timeout. Amount of time the terminal should wait to receive ACK packet from POS (in milliseconds)	User Defined	1000 ms
BAUD	Baud rate used to communicate with the POS	9600, 19200, 38400, 57600, 115200	19200
COMPRT	Comm port used to connect the POS to the FlexPoint terminal	1 - COM1 2 - COM2 3 - Ethernet 4 - Inter-App	COM1
VERBATIM_RECEIPT	When set to 'On', FlexPoint will send a verbatim transaction receipt in fields 11998 and 11999 of the response message.	0 - Off 1 - On	Off
PRINT_RECEIPT	Specifies which (if any) payment transaction receipts a FlexPoint device should print (not applicable to pin pads)	0 - both 1 - cardholder 2 - merchant 3 - none	0
SCREEN_LOCK	A security access function requiring entry of a common password	0 - Off 1 - On	Off

Config Label	Description	Values	Default
	shared by all authorized users of FlexPoint. Authentication of this password is performed by FlexPoint. This password may be configured in FlexPoint's admin section		
WAITER_AUTHENTICATION	Defines the sort of waiter authentication policy that FlexPoint shall enforce on the employee login screen. This is described in detail in §6.2	0 - No Authentication 1 - Operator ID 2 - Track Data 3 - Track or Operator ID 4 - Track and Operator ID	2
WAITER_AUTHENTICATION_TIMEOUT	Length of time to wait between the time FlexPoint receives Message ID 881 from the POS, and the waiter's selection of the ticket and payment method (which creates Message ID 89). If this time is exceeded, FlexPoint should lock, and the waiter should be required to authenticate again.	60 sec.	
IP_ADDRESS	IP address of FlexPoint unit		
HOST_NAME	Should be set to the Terminal ID of FlexPoint		

Figure 10-1 Configuration Variables

# Appendix A - AID's

To identify the card type by the associated AID's please refer to the below table for the Registered Application Identifier (RID).

Card Type	AID (RID + PIX)	RID	PIX
Amex	A00000002501	A000000025	01
Diners	A0000001523010	A000000152	3010
Discover	A0000003241010	A000000324	1010
Interac	A0000002771010	A000000277	1010
JCB	A0000000651010	A000000065	1010
MasterCard Credit	A0000000041010	A000000004	1010
MasterCard International Maestro	A0000000043060	A000000004	3060
MasterCard US Maestro	A0000000042203	A000000004	2203
Visa Credit	A0000000031010	A000000003	1010
Visa Debit International	A0000000031010	A000000003	1010
Visa Electron	A0000000032010	A000000003	2010
Visa Interlink	A0000000033010	A000000003	3010

The AID that is returned will include the RID + PIX (Proprietary Application Identifier Extension). The PIX is either a 2 or 4 digit number that is appended to the RID, for example MasterCard Credit PIX would be 1010 so the full AID would be A0000000041010. By parsing the AID you will be able to tell the card type such as Amex, Visa, Interac so you can identify that for the POS system.

- Registered Application Identifier (RID) - Represents the payment scheme (e.g. Visa, MasterCard, etc.)
- Proprietary Application Identifier Extension (PIX) - Represents the payment application type (i.e. credit, debit, prepaid, ATM-only, etc.)