



PAX Semi Application Protocol

V 1.00.05

www.pax.com.cn

Copyright © 2000-2021 PAX Computer
Technology (Shenzhen) Co., Ltd.

All rights reserved. No part of the contents of this document may be reproduced or transmitted in any form without the written permission of PAX Computer Technology (Shenzhen) Co., Ltd.

The information contained in this document is subject to change without notice. Although PAX Computer Technology (Shenzhen) Co., Ltd. has attempted to ensure the accuracy of the contents of this document, this document may include errors or omissions. The examples and sample programs are for illustration only and may not be suited for your purpose. You should verify the applicability of any example or sample program before placing the software into productive use.

Revision History

Date	Version	Note	Author
05/01/2020	v1.00.00	First edition	Liu jian
05/03/2020	v1.00.01	Update part of the message format	Liu jian
05/08/2020	v1.00.02	Update the preAuth, preAuthComplete, preAuthCancel request and response fields	Zeng tianhao
06/11/2020	v1.00.03	1. update checkBalance, sale, refund, preAuth, preAuthComplete, preAuthCancel request and response fields 2. remove the extra task	Zhang yuan
01/02/2021	v1.00.04	1. Update all information about transactions return message field in Chapter 4.2 2. Add Appendix section	Zhang yuan
19/03/2021	v1.00.05	Add the field type of result when it fails to return	Zeng tianhao

Contents

<u>1</u>	<u>INTRODUCTION</u>	<u>1</u>
1.1	PURPOSE	1
1.2	RANGE	1
1.3	NOUN INTERPRETATION	1
1.4	REFERENCE DOCUMENTATION	2
<u>2</u>	<u>OVERVIEW</u>	<u>3</u>
2.1	DEMAND SUMMARY	3
2.2	REQUIREMENTS' WORK FLOWCHART	3
2.2.1	THE POS TERMINAL HAS NETWORK COMMUNICATION CAPABILITY	3
2.2.2	THE POS TERMINAL HAS NO NETWORK COMMUNICATION CAPABILITY	4
<u>3</u>	<u>REQUIREMENTS' WORKING PRINCIPAL</u>	<u>5</u>
3.1	WORKING PRINCIPAL	5
<u>4</u>	<u>FUNCTIONAL REQUIREMENTS</u>	<u>8</u>
4.1	THE MESSAGE FORMAT	8
4.2	TYPE OF TRANSACTION	8
4.2.1	CHECK ACCOUNT BALANCE	8
4.2.2	SALE	9
4.2.3	REFUND	10
4.2.4	PRAUTH	11
4.2.5	PRAUTH COMPLETE	11
4.2.6	PRAUTH CANCEL	12
<u>APPENDIX</u>		<u>15</u>
APPENDIX–	MESSAGE FIELD	15

Table List

Table 1-1 Acronym definition	1
------------------------------	---

1 Introduction

1.1 purpose

Provide a set of reference request and response message conventions for scenarios where class ECR matches POS terminals. The ECR and POS terminals will be used to represent the two ends of such usage scenarios without special instructions.

1.2 Range

Based on current requirements, this protocol only considers scenarios where the ECR terminal initiates a request to the POS terminal. The entire transaction logic is processed on the POS terminal. When the POS terminal has network communication capability, the transaction logic occurs between the POS terminal and the Bank side. The POS terminal returns the transaction result to the ECR terminal. When the POS terminal has no network communication capability, the POS terminal needs to send the transaction message to the ECR terminal. The ECR terminal sends the transaction message to the Bank side on behalf of the POS terminal and then sends the Bank side's response or the result of timeout due to no response from Bank side to the POS terminal.

This protocol does not consider the scenario where both the ECR and POS terminal need to be synchronized. This protocol does not include the agreement on encrypting and decrypting data. The data security scheme can be contracted by the applications of both ECR and POS terminal. The POS terminal returns the final transaction result to the ECR terminal after the ECR terminal initiates a request. The protocol does not define message about the successful transaction message and the unsuccessful error code and corresponding error prompt.

This protocol does not consider the multi-task mode, where the ECR terminal can make multiple task requests to the POS terminal at the same time.

1.3 Noun interpretation

The following table format can be used to list the definitions of various special terms or the original phrases of foreign initials used in this protocol.

Table 1-1 Acronym definition

Noun	Illustrate

1.4 Reference documentation

List reference documents related to the project, such as the names and links of various documents and materials quoted in this protocol.

- XXXXX
- XXXXX

2 Overview

2.1 Demand summary

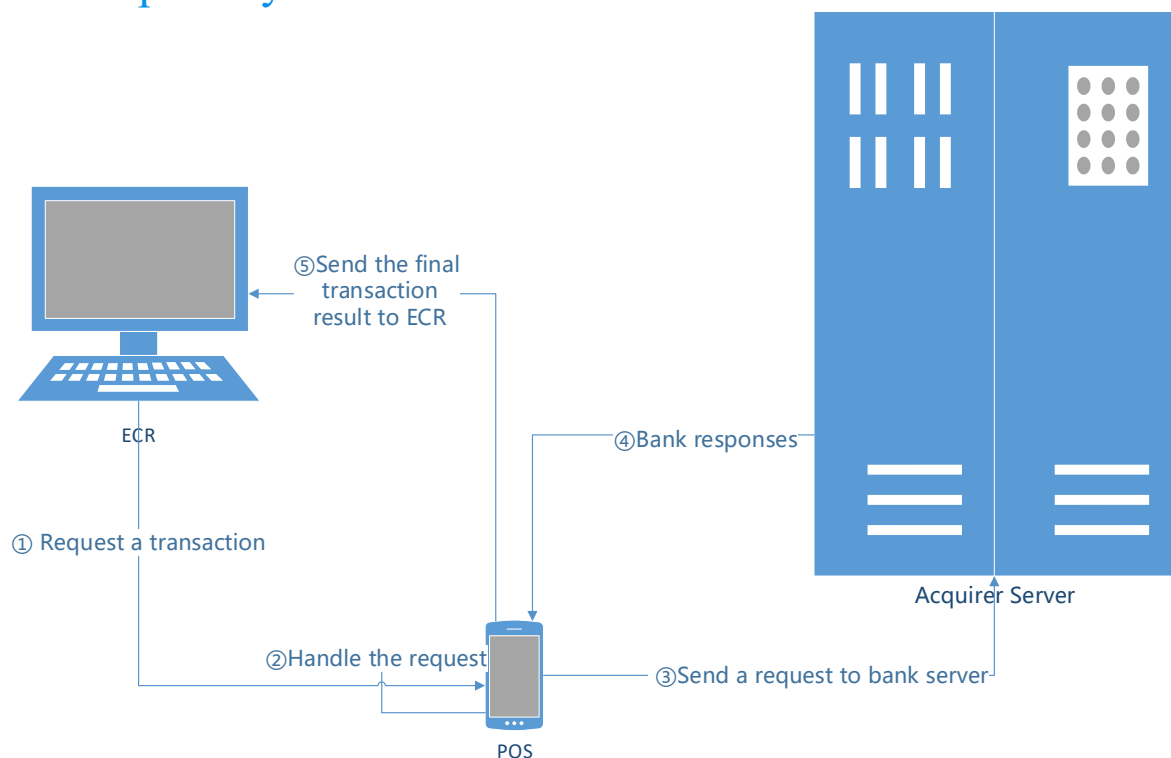
Create a unified application protocol (often referred to as the "semi Protocol") for the ECR terminal to initiate a transaction request to the POS terminal, and the entire transaction logic is processed on the POS terminal. Also, the compatibility with the Unified Open SDK should be considered.

The format of request message at ERC terminal and response message at POS terminal need to be defined in this protocol. Considering that the POS terminal may not have network communication capability, the protocol needs to define the format of request forward message and response forward message.

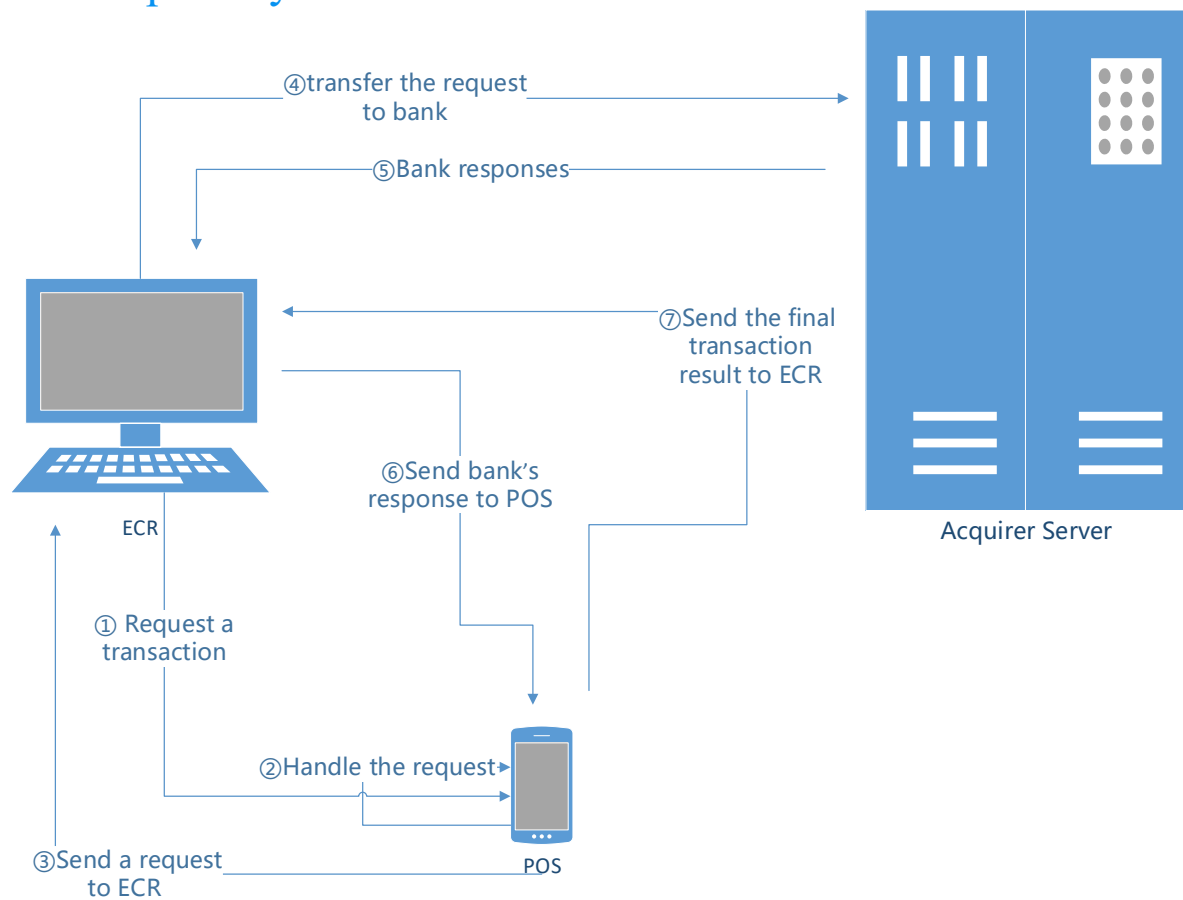
Some transaction class and management functions for reference, as well as some data fields in their request and response messages are defined in this protocol. Request message and response message are organized in JSON format.

2.2 Requirements' work flowchart

2.2.1 The POS terminal has network communication capability



2.2.2 The POS terminal has no network communication capability



3 Requirements' working principal

3.1 Working principal

The first initiator is called the requester, and the other is the responder. Follow the convention of one request, one response. The process from the beginning of a request to the end of the corresponding response is called an interaction.

According to the scope of this protocol, only one interaction occurs when the POS terminal has network capability while two interactions occur when the POS terminal has no network capability.

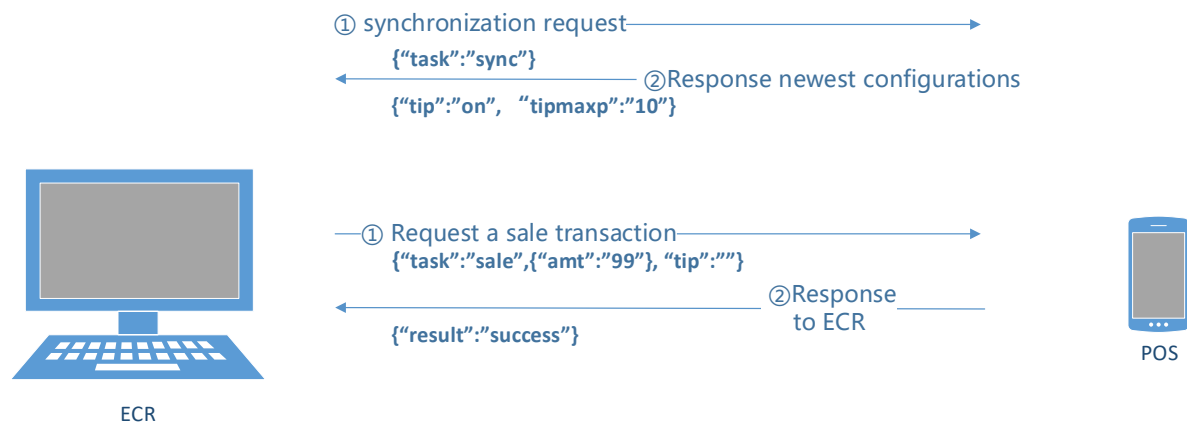
According to this protocol, the request data of the upper computer contains the task field, which indicates the task that the POS terminal is expected to perform. Some fields representing data are stipulated in this agreement, whose content is the input data or output data required for task execution.

In order to maintain the continuity between no. N ($N \geq 1$) interaction and no. $N + 1$ interaction, this protocol agreed to add a next field in the data content of the POS terminal response, which indicating that ECR terminal is expected to have subsequent requests. The content of the next field indicates the next action of the POS terminal.

Currently, the configuration of whether to support tips and tip limit percentages which needs to be updated synchronously is defined in this protocol. The upper computer needs to update the configuration to the latest state at appropriate time to ensure that its configuration information is consistent with that on the POS terminal.

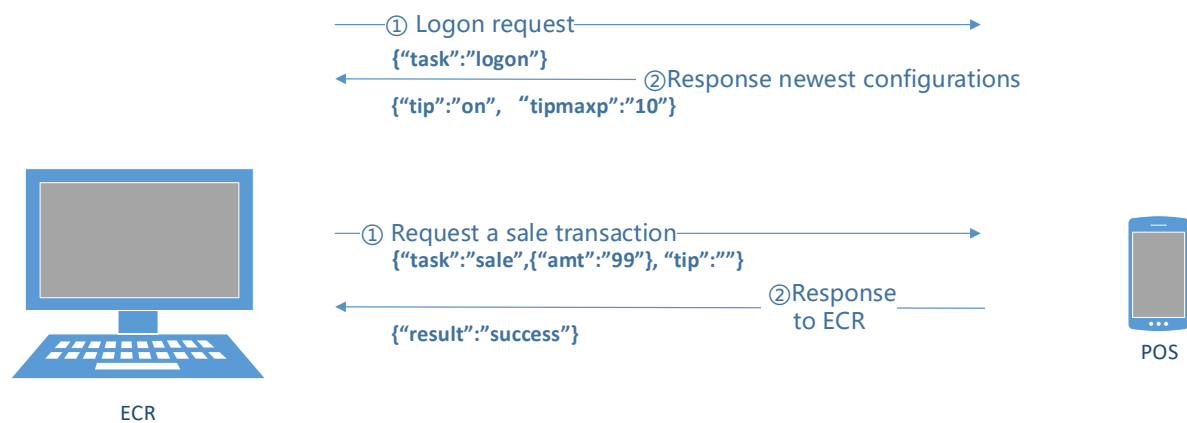
Sometimes the ECR terminal initiatively requests the POS terminal to update parameters, and then the POS terminal will inform the upper computer of the latest parameters. Sometimes, Acquirer Server will request to update parameters after the POS terminal completes the online transaction, and then the POS terminal will inform the upper computer of the latest parameters after it performs a parameter update task. When the parameters of the upper machine need to be updated depends on the specific usage scenario. The following some examples of synchronization scenarios:

1. The upper computer initiates an update request actively
2. The POS terminal has a sign-in function which the latest reference will be sent to the POS terminal with its message. The upper computer updates the parameters by initiating a sign-in request.
3. The POS terminal will receive a sign that the parameters need to update from the Acquirer Server after it completes online transaction. Then it will initiate a parameter update request to the Acquirer Server. Finally sends the latest parameters to the upper computer after the parameters are updated.



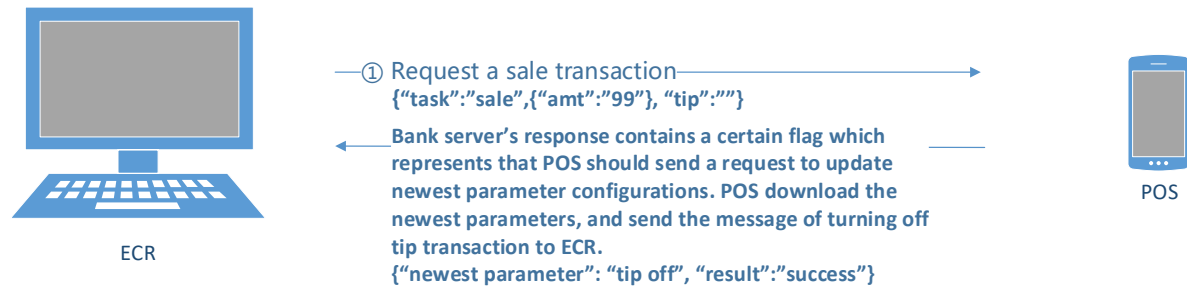
Example 1

1. ECR requests synchronization .
2. POS supports tips and maximum tip amount limit is 10% of sale amount.
3. A successful sale transaction with \$99.00 and no tips
4. sale transactions is done between POS and acquirer server.



Example 2

1. POS can get newest configurations from bank server through logon request, ECR requests logon request .
2. POS supports tips and maximum tip amount limit is 10% of sale amount.
3. A successful sale transaction with \$99.00 and no tips
4. sale transactions is done between POS and acquirer server.



Example 3

1. POS can get newest configurations when POS updates its parameter after an online transaction. .
2. POS supports tips and maximum tip amount limit is 10% of sale amount.
3. A successful sale transaction with \$99.00 and no tips
4. sale transactions is done between POS and Bank server.
5. POS update parameter configuration after sale transaction.
6. POS send the newest parameter configuration to ECR

4 Functional requirements

4.1 The message format

Messages are organized in JSON format. Two key-value pairs are defined. The value of each task's data field is different.

Key / value		Request	Response	Note to the value field of the key-value pair
Key	Value			
task	checkBalance, preAuth, preAuthComplete, preAuthCancel, sale, refund,	Have to		Choose one of the following types to request: checkBalance: Transactions to check the cardholder' account balance preAuth: Pre authorized transactions preAuthComplete: Pre-authorized completed transactions sale: Consumer transactions refund: Cancel or refund transaction
data	Filling values are different while the request types are different.	Have to	Have to	

4.2 Type of transaction

4.2.1 CHECK ACCOUNT BALANCE

4.2.1.1 Request messages

```
{
  "task": "checkBalance",
  "data": ""
}
```

4.2.1.2 Response messages

```
{
  "data": {
    "result": "transaction results",
  }
}
```

```

    "balance": "customer account balance or credit line, with decimal point,
    if there is a decimal point",
    "message": "Addition, JSON format, if not available, it can be empty"
  }

```



NOTE

The result field is used to identify the transaction results, the success represents the success of the transaction, the failure represents the transaction failure, the user timeout represents the failure of the user to timeout the operation, and the user cancel represents the user cancelled the operation.

4.2.2 SALE

The consumption amount will be immediately reflected in the cardholder's account when consumption behavior occurs on his merchant terminal.

4.2.2.1 Request messages

```

{
  "task": "sale",
  "data": {
    "amount": "Authorized amount with decimal point, such as 10.05",
    "paymentMethod": "Payment method is an optional field. If the payment
    method needs to be specified, it will be transferred to this field. If the default
    payment method is adopted, it will not be transferred to this field. For valid values,
    please refer to Message field"
  }
}

```

4.2.2.2 Response messages

```

{
  "data": {
    "result": "transaction results",
    "transId": "transaction ID",
    "message": "Addition, JSON format, if not available, it can be empty.
    Please refer to Message field",
  }
}

```



NOTE

1) The result field is used to identify the transaction results, the success represents the success of the transaction, the failure represents the transaction failure, the user timeout represents the failure of the user to timeout the operation, and the user cancel represents the user cancelled the operation.

2) The transid is used to uniquely identify the ID of the transaction which may be used for the SALE transId to query the transaction, perform a refund or cancel if a refund or cancel occurs later

4.2.3 REFUND

Cancel the original consumer transaction or return transaction.。

4.2.3.1 Request messages (EDC)

```
{
  "task": "refund",
  "data": {
    "transId": "original SALE transaction ID"
    "amount": "Refund amount with decimal point"
  }
}
```

4.2.3.2 Response messages

```
{
  "data": {
    "result": "transaction result",
    "message": "Addition, JSON format, if not available, it can be empty.
Please refer to Message field"
  }
}
```



NOTE

The result field is used to identify the transaction results, the success represents the success of the transaction, the failure represents the transaction failure, the user timeout represents the failure of the user to timeout the operation, and the user cancel represents the user cancelled the operation.

4.2.4PREAUTH

The merchant applies for the commitment of authorization amount from the card issuing bank, and the card issuing bank freezes the authorization amount of the cardholder's account and tells the merchant the authorization code.

4.2.4.1 Request messages

```
{
  "task": "preAuth",
  "data": {
    "amount": "authorized amount with decimal point"
  }
}
```

4.2.4.2 Response messages

```
{
  "data": {
    "result": "transaction result",
    "token": "preauthorized token",
    "message": "Addition, JSON format, if not available, it can be empty."
  }
}
```

Please refer to [Message field](#)



NOTE

- 1) The result field is used to identify the transaction results, the success represents the success of the transaction, the failure represents the transaction failure, the user timeout represents the failure of the user to timeout the operation, and the user cancel represents the user cancelled the operation.
- 2) The token returned by the pre-authorization represents the information item used to identify the pre-authorization in the payment application (it may be one item such as the serial number, or a combination of multiple information such as the card number + the authorization code returned from the background, etc.). When the subsequent request for the pre-authorization is completed, it will be sent back to the payment application for the pre-authorization to complete the transaction.

4.2.5PREAUTH COMPLETE

Complete the pre-authorization that was initiated in the past.

4.2.5.1 Request messages

```
{
  "task": "preAuthComplete",
  "data": {
    "amount": "actual deduction amount with decimal point",
    "token": "preauthorization returned token"
  }
}
```




NOTE

The token returned by the pre-authorization represents the information item used to identify the pre-authorization in the payment application (it may be one item such as the serial number, or a combination of multiple information such as the card number + the authorization code returned from the background, etc.). In a pre-authorized completion transaction, the token is used to find the pre-authorized transaction record (with the pre-authorized transaction record stored) and perform the pre-authorized completion, or perform the pre-authorized completion directly (without the pre-authorized transaction record stored).

4.2.5.2 Response messages

```
{
  "data": {
    "result": "transaction result",
    "transId": "transaction ID",
    "message": "Addition, JSON format, if not available, it can be empty."
  }
}
```

Please refer to [Message field](#)



NOTE

The result field is used to identify the transaction results, the success represents the success of the transaction, the failure represents the transaction failure, the user timeout represents the failure of the user to timeout the operation, and the user cancel represents the user cancelled the operation.

4.2.6 PREAUTH CANCEL

Cancel the original pre-authorization transaction.

4.2.6.1 Request messages

```
{
  "task": "preAuthCancel",
  "data": {
    "token": "preauthorization returned token"
  }
}
```



NOTE

The token returned by the pre-authorization represents the information item used to identify the pre-authorization in the payment application (it may be one item such as the serial number, or a combination of multiple information such as the card number + the authorization code returned from the background, etc.). In a pre-authorized completion transaction, the token is used to find the pre-authorized transaction record (with the pre-authorized transaction record stored) and perform the pre-authorized completion, or perform the pre-authorized completion directly (without the pre-authorized transaction record stored).

4.2.6.2 Response messages

```
{
  "data": {
    "result": "transaction result",
    "message": "Addition, JSON format, if not available, it can be empty.
Please refer to Message field"
  }
}
```



NOTE

The result field is used to identify the transaction results, the success represents the success of the transaction, the failure represents the transaction failure, the user timeout represents the failure of the user to timeout the operation, and the user cancel represents the user cancelled the operation.

4.2.6.3 请求报文

```
{
  "task": "requireData",
  "data": {
    "tag": "Tag. Specify the type of data that needs to be fetched by the payment
application" ,
    "param": "The parameter needed to get this data, can be an empty string" ,
  }
}
```

4.2.6.4 响应报文

```
{
```

```
“data”:{  
    “result” : “Transaction results” ,  
    “data” : “Acquired data” ,  
    “message” : “Additional information in JSON format, can be empty if not  
available, see Message field for details”  
}  
}
```

Note:

The result value of "success" means success, "failure" means failure, customer cancellation, customer operation timeout and other cases are treated as failure.

Appendix

Appendix– Message field

In addition to the necessary fields (result, transaction / task ID) returned by all kinds of transactions or tasks, the message field is defined to meet the special needs of some projects. In some projects, for example, it's required to return the Masked PAN, PaymentMethod, etc. for printing documents or to identify transaction information in the upper machine order management system.

The definition of data that may be returned is now uniformly defined as follows in order to ensure consistency in this protocol:

Field's name	range	Type in JSON	remark
paymentMethod	bankCard qrCode mobilePay mifareCard	string	Payment method, which is used to return the current payment method and for the upper computer to record the current payment method to its order management system, or to judge whether there is a need for follow-up operation according to the current payment method. For example, when the payment module detects a Mifarecard, the upper computer will call the relevant instructions of Mifarecard for authorization, card reading, card writing and other operations.
maskedPan		string	Masked primary account (PAN)
terminalId		string	Payment terminal ID
acquirerName		string	The name of the receiving institution for the current transaction connection
batchNo		string	Used to identify the lot number of the current transaction belongs