Research on QR Code-Based Payments and its Application in Emerging Markets

A Research Report for the Bill and Melinda Gates Foundation

October, 2017

Carol Coye Benson

Shiv Vadivelalagan

With support from Allen Weinberg & Russ Jones

GLENBROOK

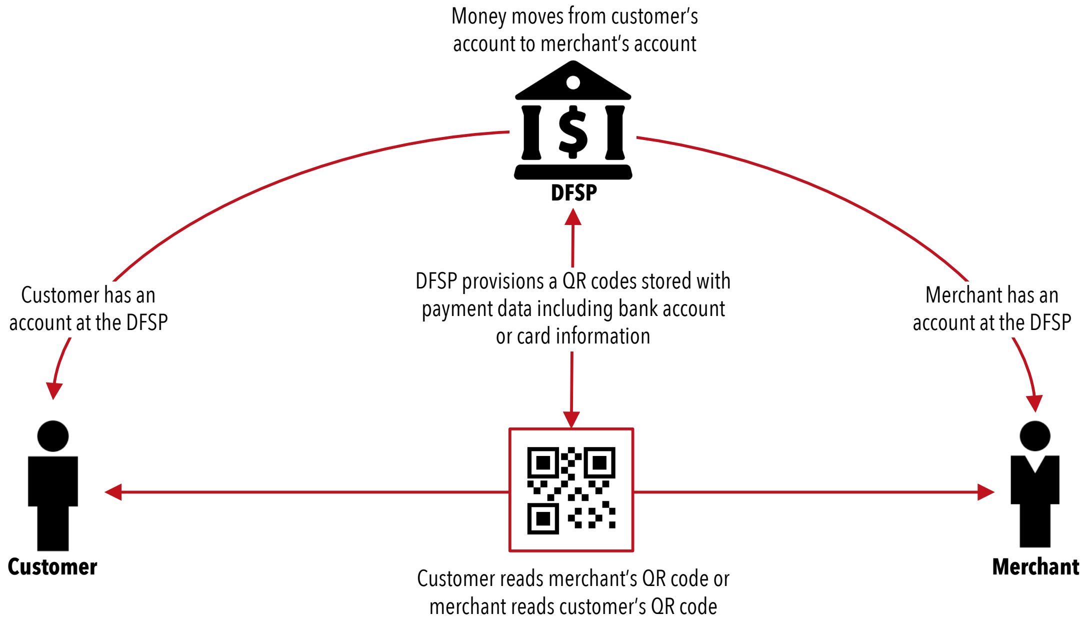
Table of Contents

Introduction

1

Introduction

A QR code can be used to store payment data and the stored data can be easily and quickly exchanged between customers (payers) and merchants or billers (payees) by scanning the QR code.


Money moves from customer's account to merchant's account
DFSP
Customer has an account at the DFSP
DFSP provisions a QR codes stored with payment data including bank account or card information
Merchant has an account at the DFSP
Customer
Merchant
Customer reads merchant's QR code or merchant reads customer's QR code

THE OBJECTIVE OF THE RESEARCH is to study how QR codes work and their application in the payments industry, and how this fits into the Level One Project framework of the Bill & Melinda Gates Foundation. The research dives into the most promising QR code-based payment models, and provides a framework that will capture critical elements to define and understand QR code-based payment services. The research documents topics of special interest such as interoperability and risks in QR code-based payments. Finally, the research presents the emerging trends in QR code-based payments and Level One perspective on QR code-based payments.

QR Code-Based Payments and the Level One Project

Why QR code-based payments are important right now

- Converting merchant and biller payments from cash to electronic forms is probably the most important factor in increasing wallet usage and eventually attaining "digital liquidity"
- Increasingly, QR codes are being seen as "the solution" to merchant and biller payments in the emerging economies
- There are lots of options for how to do QR codes – some good, some bad from the point of view of the Level One Project design principles
- The Gates Foundation has an opportunity to influence design choices as the QR code industry advances

Key Level One Principals at Stake

Push, not pull payments

Interoperability – no provider or vendor lock-in

Transaction security

History and Origin of QR Codes

The predecessor of QR codes are one-dimensional barcodes.

A barcode is an optical, machine-readable, representation of data.

Later, two-dimensional barcodes were developed which use rectangles, dots, hexagons and other geometric patterns in two dimensions that allow for higher data storage capacity.

0 36000 29145 2

Barcodes got its name because of the usage of parallel lines (bars) to systematically represent data by varying the widths and spacing of parallel lines

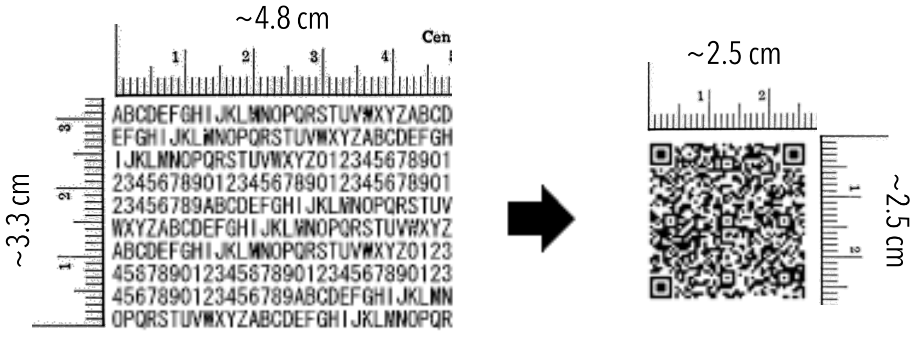
There are several types of two-dimensional barcodes: Aztec Code by Honeywell, CyberCode by Sony, Data Matrix by Microsan Systems, and MaxiCode by United Parcel Service, among others.

A QR code is a type of two-dimensional barcode which was invented and patented by Denso Wave, a subsidiary of Toyota, in 1994. Although Denso Wave retains ownership and the patent rights, it has decided not to exercise these rights. This allowed a quick uptake in the use of QR codes in various industries. Positive attributes of QR codes such as their ability to be scanned and have data transmitted in a very short span of time, also contributed to what became a surge in global usage of QR codes. ISO approved QR codes and standardized QR code symbology in 2000.

QR Codes

Quick Response (QR) Codes are ISO 18004-compliant encoding and visualization of data, which are machine-readable.

Conventional bar codes can store a maximum of approximately 20 digits but a QR Code can store several times more information.


~4.8 cm
Cen
~3.3 cm
ABCDEFGHIJKLMNOPQRSTUVWXYZABCD
EFGHIJKLMNOPQRSTUVWXYZABCDEFGH
IJKLMNOPQRSTUVWXYZ012345678901
234567890123456789012345678901
23456789ABCDEFGHIJKLMNOPQRSTUV
WXYZABCDEFGHIJKLMNOPQRSTUVWXYZ
ABCDEFGHIJKLMNOPQRSTUVWXYZ0123
456789012345678901234567890123
456789012345678 9ABCDEFGHIJKLMN
OPQRSTUVWXYZABCDEFGHIJKLMNOPQR
~2.5 cm
~2.5 cm

A QR Code symbol of this size can encode 300 alphanumeric characters.

Although QR codes were invented for use in the automobile industry, QR codes are today used in several industries including payments industry. We attribute this to due to the fast readability and greater storage capacity of QR codes compared to conventional bar codes.

TYPES OF QR CODES

QR CODE MODEL 1 and 2

Model 1 is the original QR code and Model 2 is the improvised version with greater storage capacity. The term 'QR code' generally refers to this category.

MICRO QR CODE

Smaller in size and therefore requires less printing area. However, micro QR codes can store less information than other categories.

iQR CODE

iQR codes can be printed in rectangular module; can hold 80% more information than Model 1 and 2; and have a higher data restoration capability if the code is dirty or damaged.

SQRC

Comes with reading restriction function; can be read only by certain type of readers; has the ability to encode 2 levels of data – public and private; visually no different than Model 1and2.

Frame QR

Frame QR code has a "canvas area" which can be used for inserting logos, and helps in promotion, authenticity etc.

GLENBROOK

How Do QR Codes Work?

ENCODING

- The required data (or the payload*) is converted into binary data and it is stored in the small dots in the "data area" of the QR code module
- In a QR code, data is encoded in both horizontal and vertical components of the image
- A QR code uses four standardized encoding modes (numeric, alphanumeric, byte/binary, and kanji) to efficiently store data
- Sometimes the required data is formatted using a standard formatting rule to ensure consistent data interpretation and processing. In that case, the formatted data becomes the payload

*The payload is the part of transmitted data that is the actual intended message

READING

- QR codes can be read using any imaging device such as a cellphone camera
- Another advantage of QR codes is that they are capable of omni-rotational reading (can be read in any direction in 360º)
- The imaging device locates the three distinctive squares at the corners of the QR code image and uses the smaller square(s) near the fourth corner to achieve 360º high speed reading and to negate background disturbances

DECODING

- Decoding of QR code is done by a decoding algorithm
- The decoding algorithm is stored in a programmed processor
- The processor digitally analyzes the small dots in the "data area" of the QR code module and decodes the QR code symbol into payload
- A QR code also has an error-correction algorithm which is used for retrieving the data if the code is dirty or damaged

PARSING, VALIDATING & PROCESSING

- The decoded payload is parsed and validated by the processor to retrieve the required data
- If the required data is not formatted using a standard rule, the programmed processor will not be able to retrieve the required data from the payload
- If the required data is not retrievable, further transaction processing is not possible

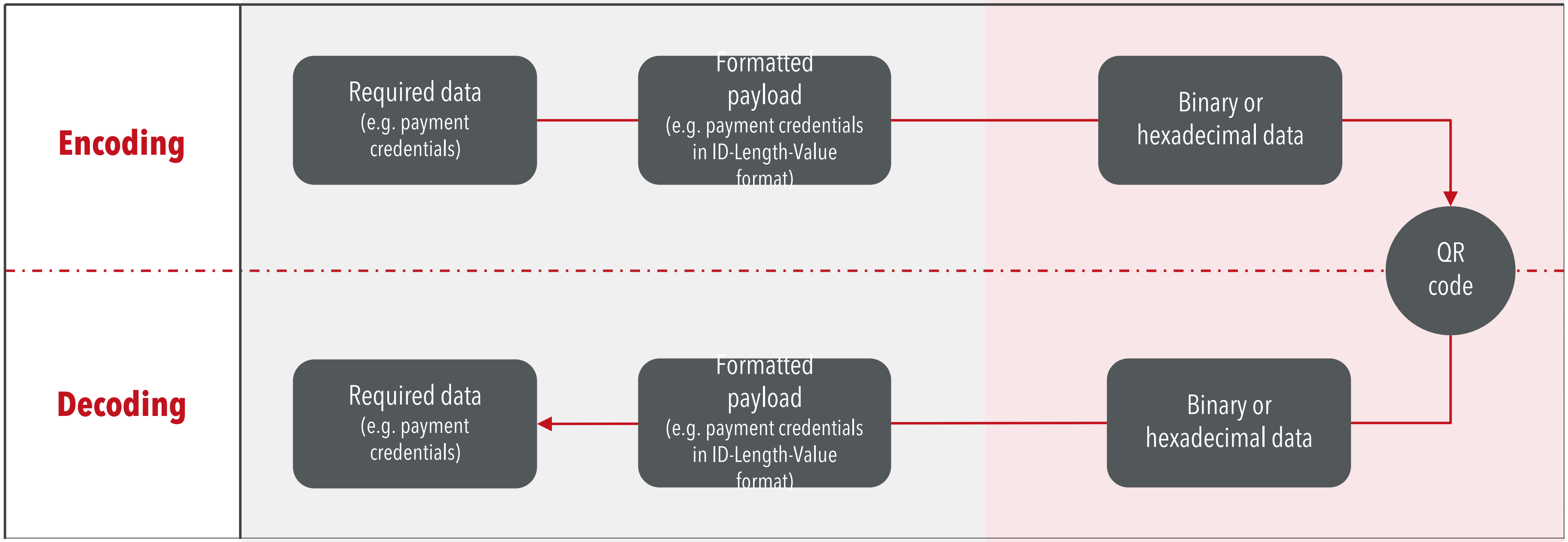
Two Levels of Data Encoding/Decoding

LEVEL 1: DATA FORMATTING

Various data objects in the required data is formatted using a standard formatting rule such as ID/type-length-value. Formatting of data objects is helpful for standardization of data interpretation and data processing when data gets exchanged across multiple entities

LEVEL 2: DATA ENCODING

Convert the formatted payload into binary or hexadecimal data format. Encode the data into a QR code symbol using one of the encodation modes per ISO 18004



Encoding

Required data (e.g. payment credentials)

Formatted payload (e.g. payment credentials in ID-Length-Value format)

Binary or hexadecimal data

QR code

Decoding

Binary or hexadecimal data

Formatted payload (e.g. payment credentials in ID-Length-Value format)

Required data (e.g. payment credentials)

Parse and validate the decoded payload using the same formatting rule used while encoding, and retrieve the required data

Decode QR code back into binary or hexadecimal data using ISO 18004 decodation specifications. Convert the binary or hexadecimal data back into formatted payload

QR Code-Based Payments

2

QR Code-Based Payments: Key Choices

FUNDAMENTAL CHOICES

There are three sets of fundamental choices in the use of QR codes in the payments industry. These are being combined in a variety of ways. Some combination of choices require one or both of the parties to be using a smart device.

1 **Presentation Mode**
Is the QR code presented by the merchant or consumer

2 **QR Code Type**
Is the QR code static or dynamic

3 **Payment Type**
Is the payment Push or Pull

OTHER FACTORS TO CONSIDER

- **Payment Data**: What payment data objects to encode in QR codes and how to format it
- **Use Cases**: What payment uses cases to enable using QR codes
- **Interoperability**: Whether the QR code provisioned by one DFSP is compatible with other DFSPs in the market

QR Code-Based Payments: Three Important Steps

Regardless of the choices made, there are three important steps in enabling a payment via QR code.

ENCODING

DFSPs format the payment credentials using a standard formatting rule, convert it into a data payload, and encode the payload into a QR code

PROVISIONING

DFSPs provision the QR code to the end user (consumer or merchant) who then presents it to the counter party for scanning

RETRIEVAL & PROCESSING

A mobile payment app with QR code reader captures the QR code, retrieves the data payload, and initiates a payment transaction

Fundamental Choices

3

Static and Dynamic QR Codes

Static QR Code

A QR code is considered static when the same QR Code is shown for more than one transaction.

A static QR code generally contains payment credentials such as PAN or payments address.

Sometimes a static QR code is unique to a user. For instance, payee (biller) assigns a unique QR code to a payer (customer) with static information such as biller's payments address, customer's account number and recurring bill amount.

Dynamic QR Code

A QR code is considered dynamic when a new QR code is shown for each transaction.

Dynamic QR codes are used to store different types of information:

1. Transaction specific information: Merchants create a dynamic QR code to capture transaction specific information such as order reference #, bill amount, etc.

2. Tumbling payment credentials: Issuers create a dynamic QR code to store one-time or limited-use cryptogram in customer presented models

QR Code-Based Payments Possibilities

STATIC QR CODES

	PUSH or REQUEST-TO-PUSH	PULL
Merchant presents QR code (and customer scans)	**PUSH** • Customer scans the QR code • Merchant passes payments address to customer via QR code • Customer enters transaction amount in their app • Customer initiates a Push payment • *E.g. Paytm, Alipay, WeChat Pay, Masterpass QR*	**PULL**
Customer presents QR code (and merchant scans)	**REQUEST-TO-PUSH** • Merchant prepares an order and scans customer QR code • Customer passes payments address to the merchant via QR code • Merchant sends a "request-to-push" message • Customer initiates a Push payment • *No real world example*	**PULL** • Merchant prepares an order and scans customer QR code • Customer passes payment credentials to the merchant via QR code • Merchant initiates a Pull payment • *E.g. Starbucks QR payment, LevelUp*

Pros and Cons Analysis

STATIC QR CODES

	PUSH or REQUEST-TO-PUSH	PULL
Merchant presents QR code (and customer scans)	**PUSH** + Reduces fixed cost of acceptance for merchants as no special hardware (e.g. smart phone) is needed – Can not capture any transaction-specific data in the merchant's systems – Merchant's QR code could be compromised to divert payments	**PULL**
Customer presents QR code (and merchant scans)	**REQUEST-TO-PUSH** + Two step verification of transaction amount (e.g. merchant enters amount and customer approves) + Can capture transaction-specific data in the merchant's systems – Expensive to deploy. Both merchants and customers need hardware (e.g. smart phone)	**PULL** + Can capture transaction-specific data in the merchant's systems – Customer's QR code could be stolen/cloned and misused – Risks associated with Pull payments are applicable

QR Code-Based Payments Possibilities

DYNAMIC QR CODES

	PUSH or REQUEST-TO-PUSH	PULL
Merchant presents QR code (and customer scans)	REQUEST-TO-PUSH • Merchant prepares an order and generates a new QR code; Customer scans the QR code • Merchant passes payments address including transaction-specific info to customer via QR code • Customers initiates a Push payment • *E.g. UPI@POS, EMVCo QR, mVISA, BharatQR,*	PULL • Merchant prepares an order and generates a new QR code; Customer scans the QR code • Merchant passes order reference # to the customer via QR code • Customer's app hands order reference # and payment credentials back to the merchant • Merchant initiates a Pull payment • *E.g. Walmart Pay*
Customer presents QR code (and merchant scans)	REQUEST-TO-PUSH	PULL • Merchant prepares an order and scans customer QR code • Customer passes dynamic/tokenized payment credentials to the merchant via QR code • Merchant initiates a Pull payment • *E.g. Alipay (tumbling limited-use QR code), EMVCo QR and Chase Pay (with dynamic cryptogram for one-time use)*

Pros and Cons Analysis

DYNAMIC QR CODES

	PUSH or REQUEST-TO-PUSH	PULL
Merchant presents QR code (and customer scans)	**REQUEST-TO-PUSH** + Two step verification of transaction amount + Can capture transaction-specific data in the merchant's systems + Secure as new QR code is generated for each transaction – Expensive to deploy. Both merchants and customers need hardware (e.g. smart phone)	**PULL** + Two step verification of transaction amount + Can capture transaction-specific data in the merchant's systems – Expensive to deploy. Both merchants and customers need hardware (e.g. smart phone) – Risks associated with Pull payments are applicable
Customer presents QR code (and merchant scans)	**REQUEST-TO-PUSH**	**PULL** + Can capture transaction-specific data in the merchant's systems + Secure as new QR code is generated for each txn – Expensive to deploy. Both merchants and customers need hardware (e.g. smart phone) – Risks associated with Pull payments are applicable

A Level One Perspective on QR Code Choices

Static vs. Dynamic

DFSPs should offer both static and dynamic codes to merchants in a market

- Static QR codes reduce upfront costs and therefore it is more suited for enabling payments among poor merchants and low-income customers
- Static QR codes are more vulnerable to frauds. Therefore, static QR codes must be used to enable Push payments only, and the merchant should receive a payment receipt confirmation message via SMS or other media
- Dynamic codes are more secure and drives more utility to merchants

Customer presented vs. Merchant presented

Merchant presented QR code is preferable from a financial inclusion view point

- It is easier to enable a Push payment when merchant (or payee) presents the QR code
- Request-to-Push payments can be enabled when customer (or payer) presents the QR code. However, this arrangement requires both customers and merchants to have smart devices
- Customer presented static QR codes supporting Pull payments shall be avoided

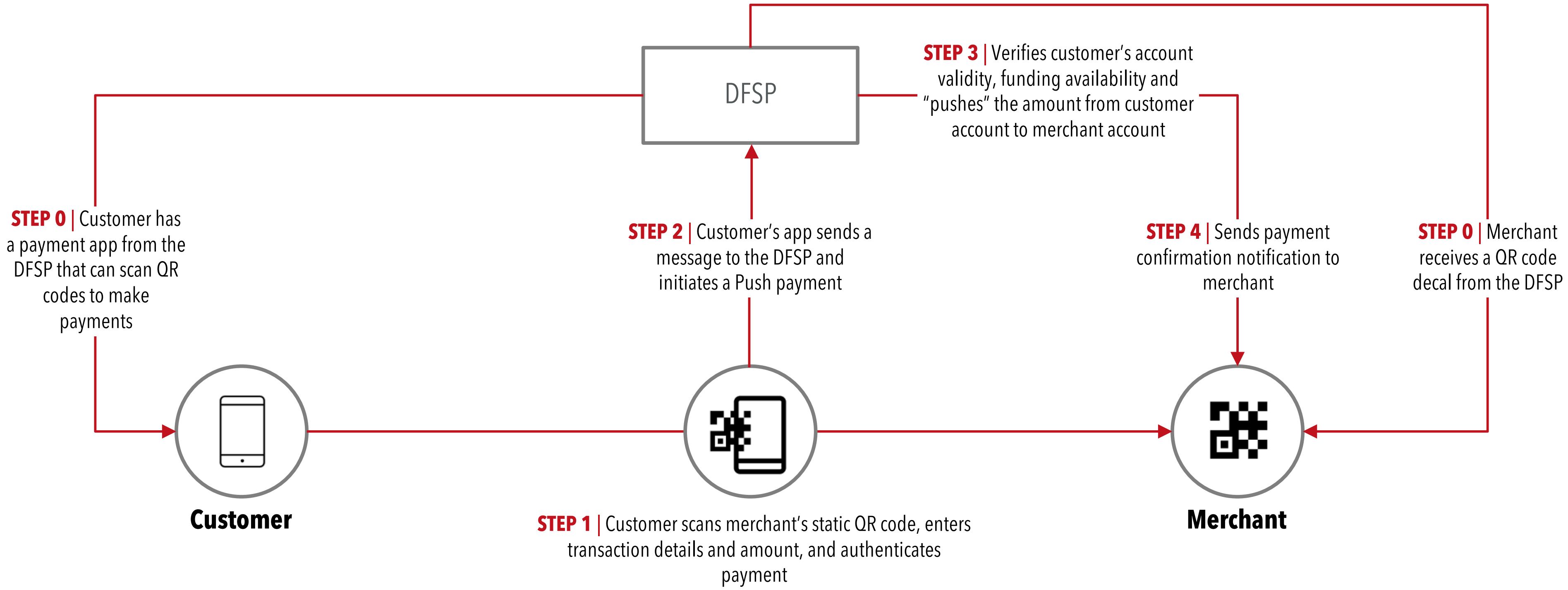
Push vs. Pull payment

Level One advocates for Push payment or request-to-Push

- Level One supports QR code arrangements that are suited for Push payments such as merchant presented QR code solution (dynamic and static with payment receipt confirmation)

Transaction Flow

Merchant presents a static QR code [Push payment]



STEP 0 | Customer has a payment app from the DFSP that can scan QR codes to make payments

STEP 0 | Merchant receives a QR code decal from the DFSP

STEP 1 | Customer scans merchant's static QR code, enters transaction details and amount, and authenticates payment

STEP 2 | Customer's app sends a message to the DFSP and initiates a Push payment

STEP 3 | Verifies customer's account validity, funding availability and "pushes" the amount from customer account to merchant account

STEP 4 | Sends payment confirmation notification to merchant

EXAMPLES

支付宝 ALIPAY

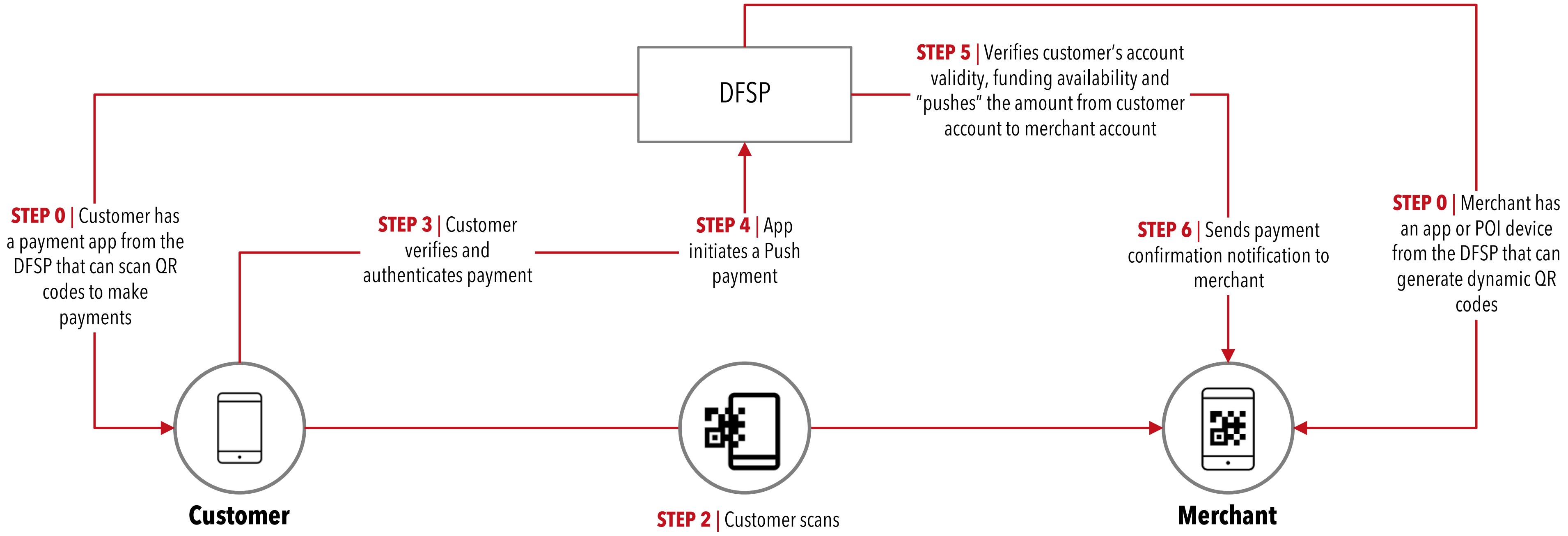
masterpass

WeChat Pay

Paytm

Transaction Flow

Merchant presents a dynamic QR code [Request-to-Push payment]



STEP 0 | Customer has a payment app from the DFSP that can scan QR codes to make payments

STEP 0 | Merchant has an app or POI device from the DFSP that can generate dynamic QR codes

STEP 1 | Merchant prepares an order, and generates a dynamic (transaction-specific) QR code

STEP 2 | Customer scans merchant's dynamic QR code

STEP 3 | Customer verifies and authenticates payment

STEP 4 | App initiates a Push payment

STEP 5 | Verifies customer's account validity, funding availability and "pushes" the amount from customer account to merchant account

STEP 6 | Sends payment confirmation notification to merchant

EXAMPLES

UPI @ POS
UNIFIED PAYMENTS INTERFACE

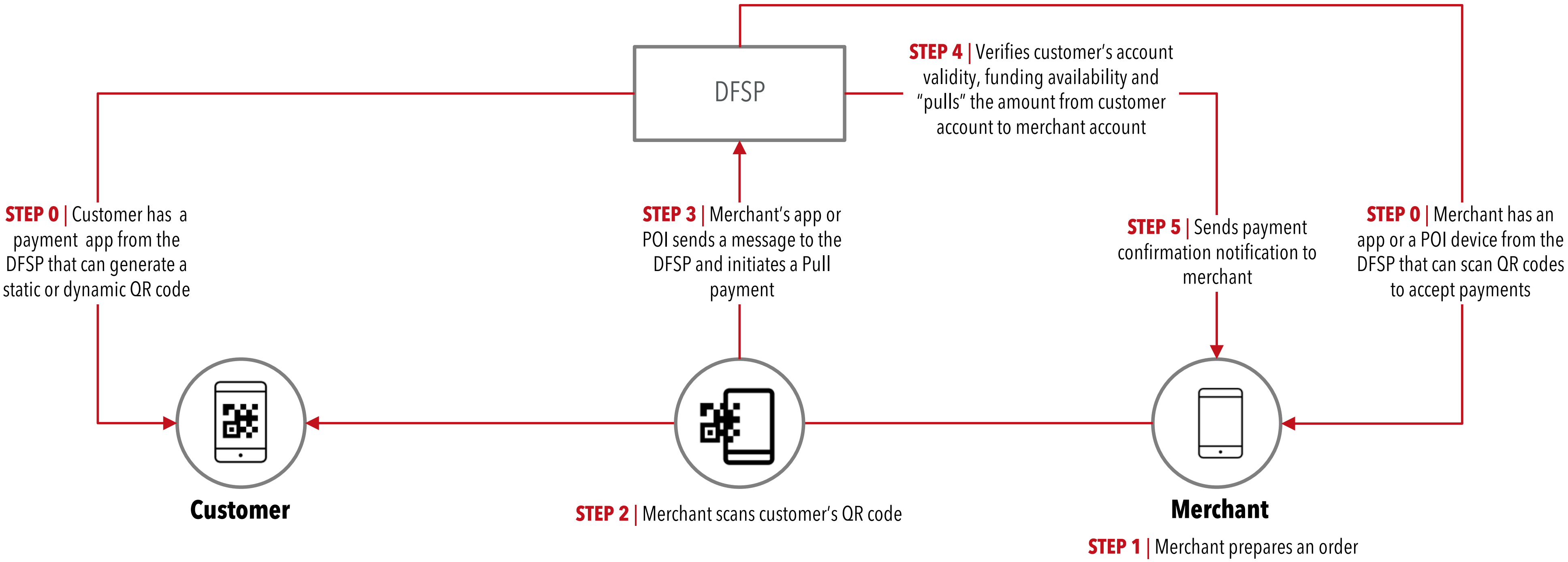
QR Codes
The EMV® QR Code Specifications

mVISA

BHARATQR

Transaction Flow

Customer presents a static or dynamic QR code [Pull payment]



DFSP

STEP 0 | Customer has a payment app from the DFSP that can generate a static or dynamic QR code

STEP 0 | Merchant has an app or a POI device from the DFSP that can scan QR codes to accept payments

STEP 1 | Merchant prepares an order

STEP 2 | Merchant scans customer's QR code

STEP 3 | Merchant's app or POI sends a message to the DFSP and initiates a Pull payment

STEP 4 | Verifies customer's account validity, funding availability and "pulls" the amount from customer account to merchant account

STEP 5 | Sends payment confirmation notification to merchant

EXAMPLES

支付宝 ALIPAY

Pay℠

LevelUp

STARBUCKS COFFEE

Other Considerations

4

Data Objects in QR Codes Used for Payments

The data objects in QR code used for payments include payment credentials such as payments address, account number, payment token or DFSP ID, and sometimes also include transaction-specific information such as transaction amount, tip/ convenience fee indicator, etc. The exact data objects in a payment data depend on the following two factors, and of course, what is allowed by the QR code provisioning DFSP.

1. Who is presenting the QR code – customer (payer) or merchant (payee)
2. Whether the QR code is static or dynamic

	QR Code is static	**QR Code is dynamic**
Merchant presents QR code (and customer scans)	Merchant account information DFSP identification number Payment Network ID Etc.	Merchant account information Customer account information* DFSP identification number Payment network ID Order reference number Transaction amount Tip indicator Etc.
Customer presents QR code (and merchant scans)	Customer account information DFSP identification number Payment Network ID Etc.	Customer account information Dynamic Cryptogram DFSP identification number Payment Network ID Etc.

GLENBROOK

*Example: Walmart Pay

Data Formats Used in QR Code-Based Payments

The DFSPs need to create or choose a standard to format the required data (e.g. payment credentials) that will specify data objects that are mandatory and optional, tag/ID of data objects, data objects organization, allowed length and value and content of data objects. The data objects could be placed in a URI format or in a structured data format to construct a payment message

EXAMPLE

URI formatting

- A simple formatting approach in which payment data elements are formatted into a URI which is encoded in the QR code
- On scanning using a mobile payment app, the URI prefills payment details in the app
- The URI can be interpreted by the mobile payment app only

UPI UNIFIED PAYMENTS INTERFACE

scan2pay

European Payments Council

Standard European QR code

Application-level formatting

- A sophisticated formatting approach in which payment data objects are formatted using a specific formatting rule
- On scanning, the app parses and validates the data objects, and constructs a payment message
- Must be scanned using a specific mobile payment app designed to retrieve the data objects

Pay

QR Codes

The EMV® QR Code Specifications

mVISA

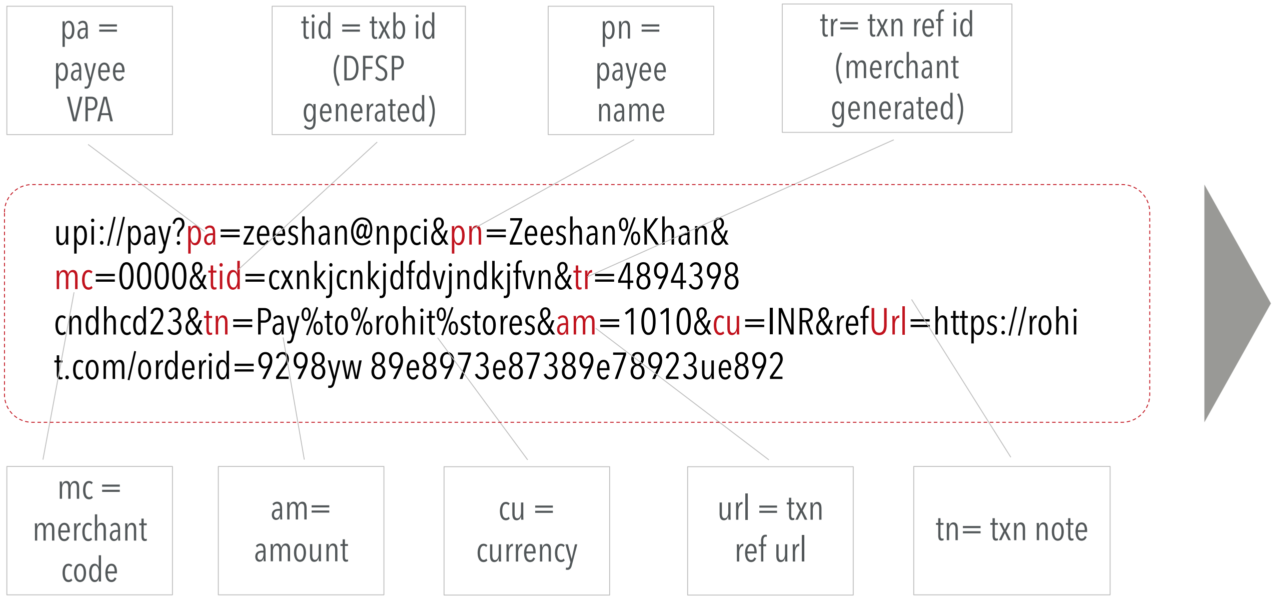
Worked Example: Two Types of Data Formatting

URI Formatting

Use case: Customer makes a payment to merchant at a store

Procedure: Merchant uses a mobile payment app to enter details of the transaction. The app formats the payment data into an URI (data payload), which gets coded into a QR code

Example: UPI (India) merchant presented dynamic QR code



pa = payee VPA

tid = txb id (DFSP generated)

pn = payee name

tr= txn ref id (merchant generated)

upi://pay?pa=zeeshan@npci&pn=Zeeshan%Khan&
mc=0000&tid=cxnkjcnkjdfdvjndkjfvn&tr=4894398
cndhcd23&tn=Pay%to%rohit%stores&am=1010&cu=INR&refUrl=https://rohi
t.com/orderid=9298yw 89e8973e87389e78923ue892

mc = merchant code

am= amount

cu = currency

url = txn ref url

tn= txn note

When customers scans the QR code using the mobile payment app, the app prefills the details. Customer confirms the details, and complete the payment

Source: UPI: Common URL specifications for deep linking and proximity integration

GLENBROOK

Worked Example: Two Types of Data Formatting

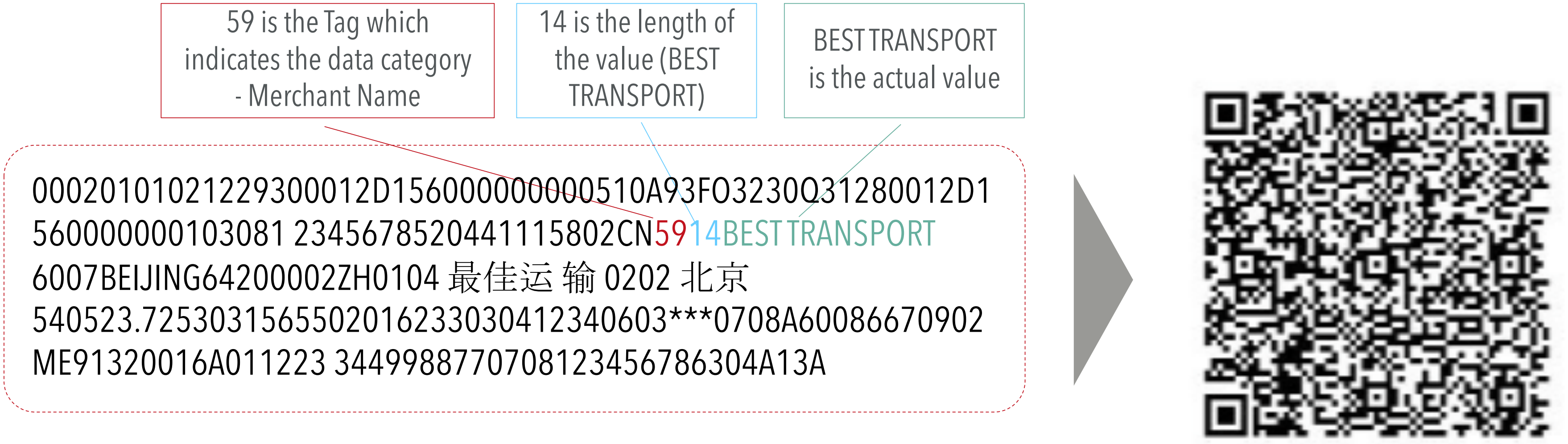
Application Level Formatting

Use case: Customer makes a payment to merchant at a store

Procedure: Merchant uses a mobile payment app to enter details of the transaction. The app formats the payment data using a standardized formatting rule, coverts it into a data payload, and encode the payload into a QR code

Example: EMVCo merchant presented dynamic QR code

EMVCo prescribes that payment data is formatted using Tag-Length-Value (TLV) formatting rule.



59 is the Tag which indicates the data category - Merchant Name

14 is the length of the value (BEST TRANSPORT)

BEST TRANSPORT is the actual value

00020101021229300012D156000000000510A93FO3230Q31280012D1
560000000103081 2345678520441115802CN5914BEST TRANSPORT
6007BEIJING64200002ZH0104 最佳运 输 0202 北京
540523.72530315655020162330304123406 03***0708A60086670902
ME91320016A011223 3449988770708123456786304A13A

When customer scans the QR code using the mobile payment app, the app will initiate the payment transaction

Source: EMV QRCPS

Use Cases Enabled by QR Code-Based Payments

QR codes can enable a number of payment use cases other than Point of Sale (POS)

Use Cases	Remote Commerce	Bill Payment	P2P Person to Person	B2B Business to Business
How does it work?	Similar to a POS transaction, customer scans a transaction-specific dynamic QR code on the checkout page and authenticates payment.	QR code encoded with biller direct on the statement is scanned by any generic QR code reader which redirects customer to the biller's web page with pre-filled billing info and amount. Customer manually enters payment details and authenticates. Or, QR code on bill statement is scanned by a payment app which takes the customer to the payment page in the app with pre-filled billing info and amount . User just authenticates the payment.	Payer scans payee's QR code, enters amount and authenticates payment. Payee creates a "request-to-Push" by entering amount and generates a dynamic QR code. Payer scans the transaction-specific QR code and authenticates the payment.	QR code is included on B2B supplier invoice, and upon delivery, buyer scan the QR code which loads a payment page with pre-filled supplier information. Buyer initiates payment via one of the payment methods such as ACH, debit card or credit card
Example	• SnapScan, South Africa	• NACHA QR Bill, USA • BPAY QR Codes, AU • Boleto, Brazil	• Alipay & WeChat, China • Paytm, India • UPI (via BHIM), India	• QR Invoice, USA

Interoperable QR Code-Based Payments

WHAT IS AN INTEROPERABLE QR CODE?

An interoperable QR code is a standards based code that is compatible with other QR code-based payment services in the market. In other words, the payment data in an interoperable QR can be retrieved by other QR code-based payment services in the market to initiate a payment. Example: Bharat QR, India.

WHAT DOES IT ENABLE?

Interoperable QR codes enable payment transactions among users (e.g. a merchant and customer) belonging to two different DFSPs.

WHAT IS A NON-INTEROPERABLE QR CODE?

A non-interoperable QR code is a closed-loop/proprietary QR code in which the payment data is encoded in such a way that it is retrievable only by the DFSP which provisioned the proprietary QR code and not by any other QR code-based payment service in the market. Example: Alipay China, Paytm India.

HOW DOES IT WORK?

Interoperable QR code-based payments are enabled by three different ways:

1. Enabled by a scheme
2. Enabled among schemes
3. Enabled by decoupling transactions

Interoperable QR Code-Based Payments

Three different ways of enabling interoperable QR code-based payments

Within scheme interoperability

- A payment scheme introduces an interoperable QR code standard to enable interoperable QR code-based payments among all DFSPs belonging to the scheme
- Example: mVISA, Masterpass, Quick Pass QR (UnionPay)

Among schemes interoperability

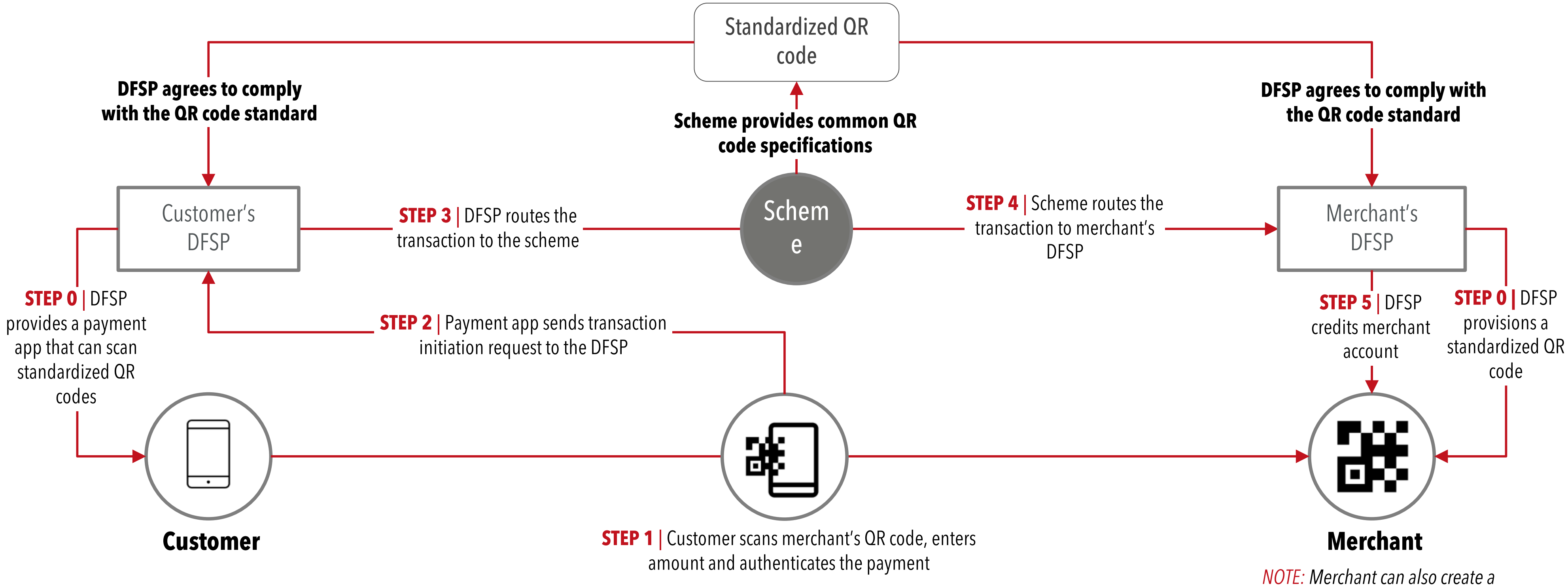
- An interoperable QR code standard introduced jointly by two or more payment schemes to enable interoperable QR code-based payments among DFSPs belonging to any of the participating schemes
- Example: EMVCo QR Code Specifications (EMVCo members), Bharat QR (Visa, Mastercard, Rupay), Thailand QR Code Standard (Visa, Mastercard, UnionPay)

Interoperability by decoupling transactions

- A payments facilitator who acts in the middle issues QR codes to merchants and has an app which consumers can use to link their credit or debit cards
- The payments facilitator originates a purchase transaction against the customer's card. When the transaction is authorized, they settle separately with the merchant using a domestic credit transfer system
- Example: SnapScan (ZA), Zapper (EU)

Within Scheme Interoperability

Merchant presents a static or dynamic QR code [Push payment]



Standardized QR code

DFSP agrees to comply with the QR code standard

Scheme provides common QR code specifications

DFSP agrees to comply with the QR code standard

Customer's DFSP

STEP 3 | DFSP routes the transaction to the scheme

Schem e

STEP 4 | Scheme routes the transaction to merchant's DFSP

Merchant's DFSP

STEP 0 | DFSP provides a payment app that can scan standardized QR codes

STEP 2 | Payment app sends transaction initiation request to the DFSP

STEP 5 | DFSP credits merchant account

STEP 0 | DFSP provisions a standardized QR code

Customer

STEP 1 | Customer scans merchant's QR code, enters amount and authenticates the payment

Merchant

NOTE: Merchant can also create a transaction-specific QR code with the same standard

EXAMPLES

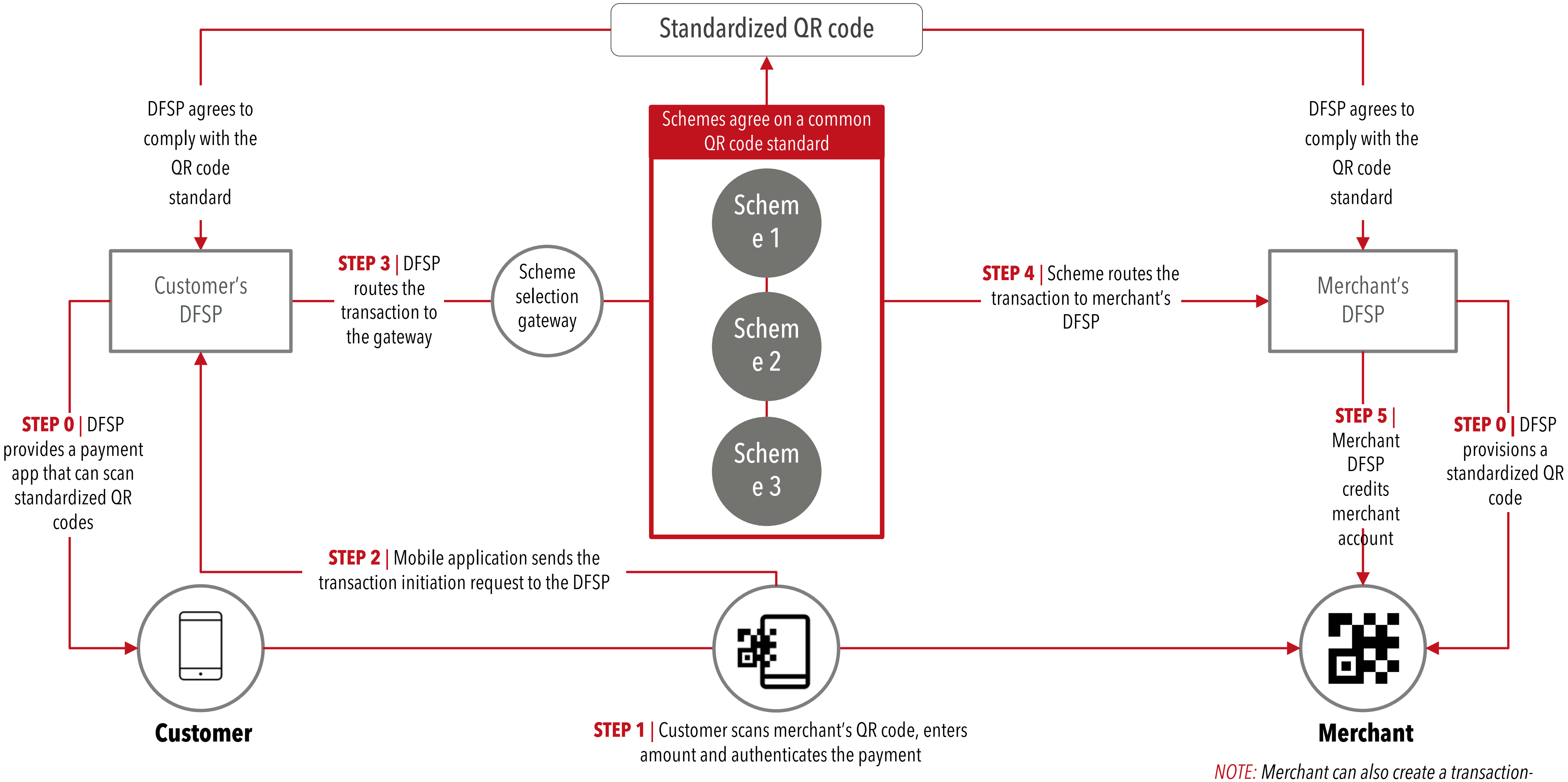
mVISA

masterpass

UnionPay 银联 UnionPay International Quick Pass QR

Among Schemes Interoperability

Merchant presents a static or dynamic QR code [Push payment]



Standardized QR code

Schemes agree on a common QR code standard

Schem e 1

Schem e 2

Schem e 3

DFSP agrees to comply with the QR code standard

Customer's DFSP

STEP 3 | DFSP routes the transaction to the gateway

Scheme selection gateway

STEP 4 | Scheme routes the transaction to merchant's DFSP

DFSP agrees to comply with the QR code standard

Merchant's DFSP

STEP 0 | DFSP provides a payment app that can scan standardized QR codes

STEP 2 | Mobile application sends the transaction initiation request to the DFSP

STEP 5 | Merchant DFSP credits merchant account

STEP 0 | DFSP provisions a standardized QR code

Customer

STEP 1 | Customer scans merchant's QR code, enters amount and authenticates the payment

Merchant

NOTE: Merchant can also create a transaction-specific QR code using the same standard

EXAMPLES

BHARATQR

QR Codes

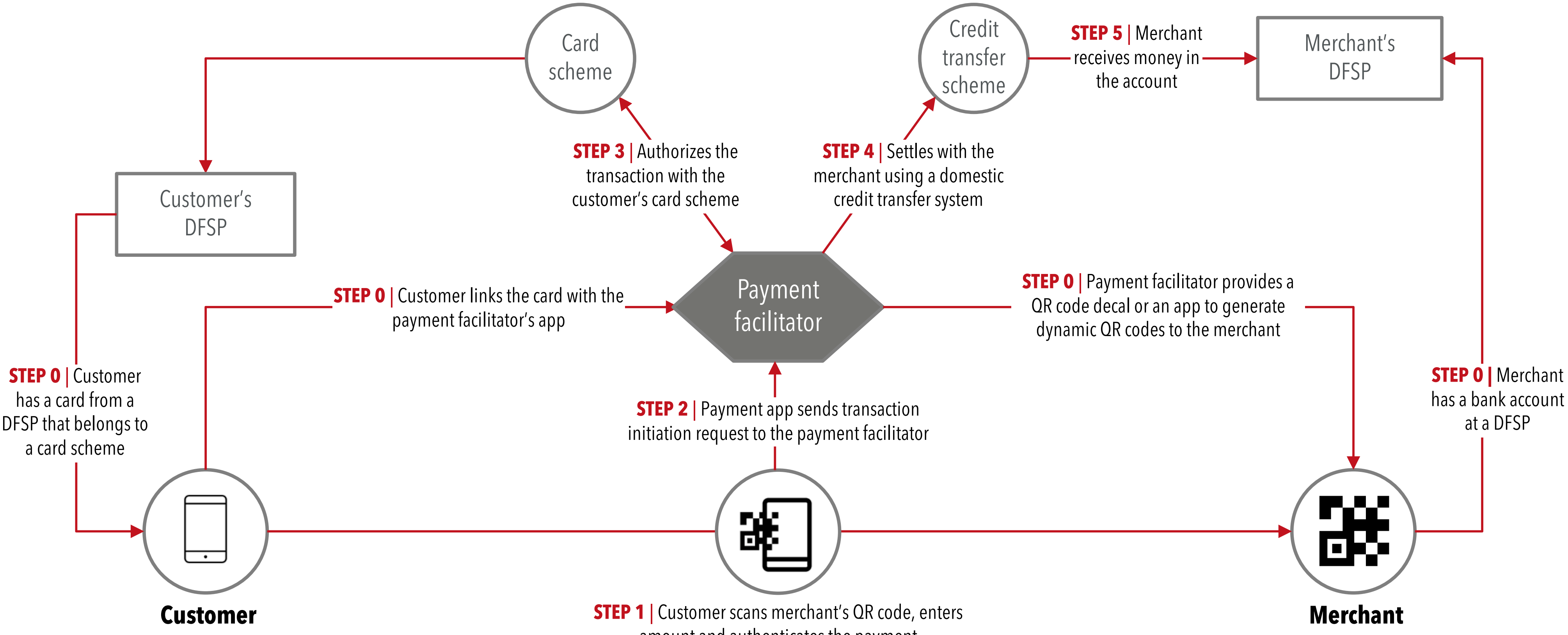
The EMV® QR Code Specifications

STANDARDIZED QR CODE

THAILAND

Interoperability By Decoupling Transactions

Merchant presents a static or dynamic QR code [Decoupled Transaction]



Card scheme

Credit transfer scheme

Merchant's DFSP

Customer's DFSP

Payment facilitator

STEP 5 | Merchant receives money in the account

STEP 3 | Authorizes the transaction with the customer's card scheme

STEP 4 | Settles with the merchant using a domestic credit transfer system

STEP 0 | Customer links the card with the payment facilitator's app

STEP 0 | Payment facilitator provides a QR code decal or an app to generate dynamic QR codes to the merchant

STEP 0 | Customer has a card from a DFSP that belongs to a card scheme

STEP 0 | Merchant has a bank account at a DFSP

STEP 2 | Payment app sends transaction initiation request to the payment facilitator

STEP 1 | Customer scans merchant's QR code, enters amount and authenticates the payment

Customer

Merchant

NOTE: Merchant can also create a transaction-specific QR code

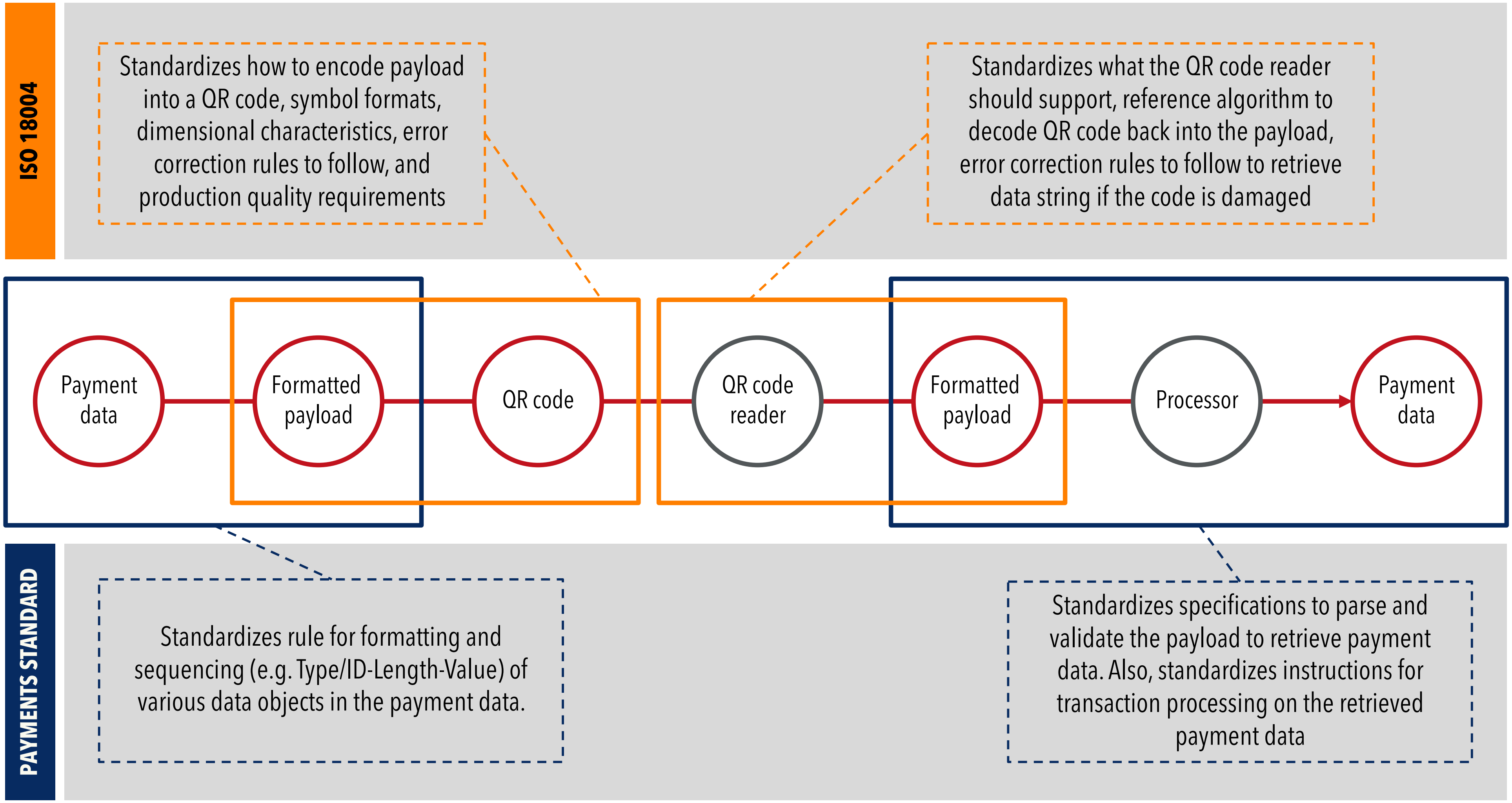
EXAMPLES

SnapScan

zapper™

Interoperable QR Code-Based Payments

Responsibility of standardization between ISO 18004 and QR code-based payments standard



Standardizes how to encode payload into a QR code, symbol formats, dimensional characteristics, error correction rules to follow, and production quality requirements

Standardizes what the QR code reader should support, reference algorithm to decode QR code back into the payload, error correction rules to follow to retrieve data string if the code is damaged

Standardizes rule for formatting and sequencing (e.g. Type/ID-Length-Value) of various data objects in the payment data.

Standardizes specifications to parse and validate the payload to retrieve payment data. Also, standardizes instructions for transaction processing on the retrieved payment data

Risks in QR Code-Based Payments

There are multiple potential risks in the evolving QR code marketplace. Establishing appropriate mitigants early on is important to preserve consumer and merchant trust in the system

Inherent risks in QR code technology

- The level of difficulty to generate a QR code is very low. Anyone can generate a QR code and encode it with malicious links
- Since the authenticity of QR codes can not be verified by human eye, customers are vulnerable to malware attacks when they scan a malicious QR code (pasted over a legitimate QR code)
- There have been reports of malicious QR codes installing malwares to take over the smartphone including the mobile banking app
- Fake QR codes can also be used for phishing attacks wherein the payer is redirected to a fake website that looks like the payee's (biller's) website to extract card information
- QR codes embedded with a payment URL are more vulnerable to this type of threat

Counterfeit payment credentials

- This is the most common type of risk where scammers replace legitimate merchant QR codes with fake QR codes to divert payments into the scammer's bank account or wallet
- Static merchant (payee) QR codes are vulnerable to this type of threat

Steal/clone payment credentials

- Scammers can also steal or clone customer (payer's) QR code which contains payment credentials
- Customer (payer) presented static QR codes that initiates a Pull payment are vulnerable to this type of threat

QR Code Risks and Mitigants

	RISK MITIGANTS
Inherent risks in QR code technology	• DFSPs should educate their customers to check the QR code before scanning to make sure that it is not a sticker placed over an original code • DFSP app should show the contents of the QR code to the user and wait for the user's consent to visit the link. This will allow customers to visually examine the decoded text before opening the link • Payment services using QR codes should use URIs not URLs, which are interpreted by mobile payment app only
Counterfeit payment credentials	• QR code provisioning DFSPs should tighten KYC norms for account opening making it difficult for scammers to open accounts and withdraw funds from accounts • DFSP apps should be able to detect whether a QR code is generated by its own system, or if the code is compromised. Usage of Message Authentication Code is recommended • The app should alert the payer when a security risk is detected and redirect them to safe zone
Steal/clone payment credentials	• DFSPs should try to avoid models that initiate a Pull payment using customer presented static QR code encoded with payment credentials

Conclusion

6

Trends And Conclusions

Trends

- Most interoperable QR code-based payment services deploy merchant presented mode and enable Push payments
  - Offer both static and dynamic code-based solutions in a market
- No observable trend in proprietary QR code-based solutions
  - Supports both presentation modes and both Push and Pull payments
- QR codes are becoming a cause of major concern in terms of security and fraud
- Analysis of risks and risk mitigation in QR code-based payments is still a nascent field

Best choices to enable the Level One Project vision for a new digital economy

- QR code-based payments interoperable among schemes
- QR code presented by the merchant enabling Push payments
- Offer both static and dynamic QR code-based payment solutions in a market
- To include poor customer without smart devices, DFSP should allow merchant till number based payments

Carol Coye Benson
carol@glenbrook.com

Glenbrook Partners
www.glenbrook.com

Shiv Vadivelalagan
shiv@glenbrook.com

GLENBROOK