January 2021

Refinement of Embedded Payment Links via User Reports

Hailun Zhu

Guiheng Zhou

Chun Fan

Fausto Araujo

Refinement of Embedded Payment Links via User Reports

ABSTRACT

Payment applications make use of QR codes, NFC tags, etc. to encode information related to particular merchant physical assets and locations. Large scale deployment of such codes is difficult and prone to human error since the payment application provider needs to access merchant information to generate codes specific to each asset and the merchant needs to deploy the codes to specific matching locations. This disclosure describes the use of codes with unique but generic embedded links that can be deployed at any physical asset. When a payment is attempted by scanning such a code, the user is requested to provide information that serves to identify the physical asset. When a sufficient number of reliable user reports are received, the generic link associated with the code is updated to include the user-provided information to convert it into an asset-specific link that can be utilized to associate payments via the payment application with the specific physical asset.

KEYWORDS

- Payment code
- Payment link
- Deep link
- Merchant location
- Asset identification
- Payment app
- Quick Response (QR) code
- Near Field Communication (NFC) tag

BACKGROUND

Embedded links such as deep links provided via a printed or displayed tag such as a quick response (QR) code or a near-field communication (NFC tag) are commonly utilized to help users navigate to specific online resources. For example, in payment applications that integrate payments such as parking spot payments, gas pump payments, etc., such a code is often provided on a physical asset such as a parking spot indicator or a gas pump. This enables the precise asset to be communicated to a backend service to facilitate payment processing.

As an example, a QR code or a NFC tag can be affixed to a gas pump as a sticker. A user of a payment app can initiate a payment by scanning the QR code or the user device reading the NFG tag. The information from the QR code/ NFC tag serves to identify the gas pump such that payments made by the user are matched to the specific pump. For example, upon receipt of a prepayment, the corresponding gas pump can be activated.

However, the use of such physical stickers in a large scale setting can face difficulties, both at the time of rollout as well as during replacement of stickers. Affixing the stickers to the correct physical asset that it encodes is a complicated process performed by the corresponding merchants and is therefore error prone. For example, a wrongly affixed sticker at a gas pump can result in a wrong gas pump (or wrong gas station or vendor) being activated when the code is scanned. Manufacture and distribution of stickers is also difficult, since stickers need to be customized to specific merchants and corresponding assets or locations where each sticker is to be affixed, for which access to such data is needed prior to distribution. Also, when physical assets are rearranged, removed, or renumbered, the corresponding sticker needs to be replaced to ensure accurate payment processing.

DESCRIPTION

This disclosure describes techniques for user-initiated discovery and mapping of link parameters embedded in stickers (e.g., QR codes, NFC tags, etc.) associated with physical assets. The techniques can be utilized by a provider of the stickers (e.g., a payment application vendor) to automatically associate specific physical assets (e.g., gas pumps, parking spots, merchant checkout location, etc.) with individual stickers by updating the embedded links based on information obtained from users during initial use of the sticker.

Per techniques of this disclosure, prior to deployment of an embedded link-based payment solution for a particular merchant, generic dynamic embedded links are generated that each include a unique identifier. Suitable machine readable codes, e.g. QR codes, NFC tags, other visual codes, etc. are mapped via 1:1 mapping to each generic link and link identifier, and are provided to the merchant. The merchant affixes the codes (e.g., stickers that include the codes) to physical asset(s) operated by the merchant that are to be associated with the payment application.

After deployment, when a user navigates to the embedded link, e.g. by scanning the QR code to initiate a payment transaction, the user is requested to enter parameters associated with the physical asset. For example, the merchant location is a gas station, the user may be requested to provide the gas station location and pump number. The asset-specific parameters provided by the user are stored along with the unique link identifier in a database that is hosted by the payment application provider. When sufficient matching parameters are received, e.g., to uniquely identify the physical asset and with sufficient reliability, the embedded link is refined to reflect the actual parameters associated with the physical asset. Subsequent scans of the particular code automatically identify the asset and associate the payment with it.

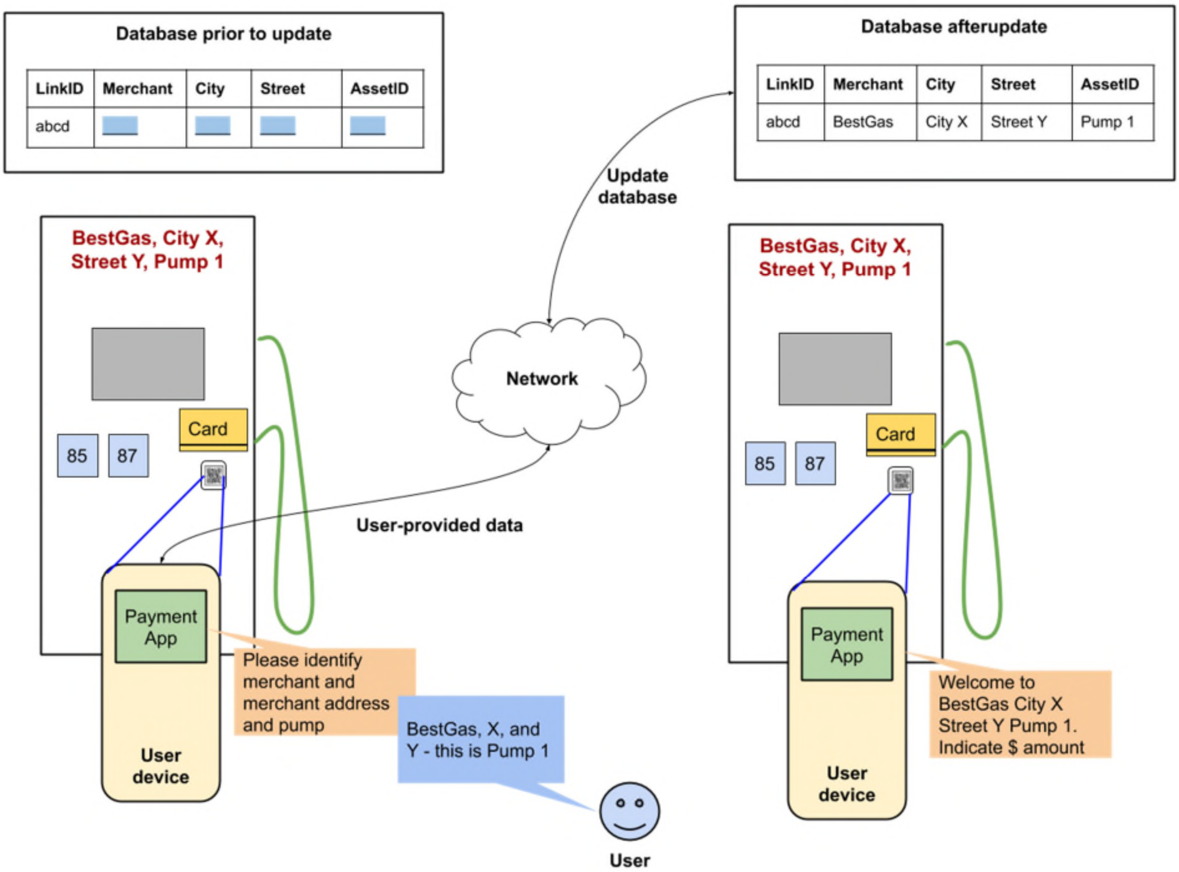

Fig. 1: Use of scannable codes to obtain asset information for payment

Fig. 1 depicts an example of updating the asset information associated with a printed code that is deployed in the field and updating the link associated with the code with appropriate parameters, per techniques of this disclosure. In this illustrative example, a gas station of merchant BestGas located at street Y of City X has multiple gas pumps, each with an associated code. In the example shown, a QR code is affixed to pump 1. A user attempts to activate the pump via a payment application on their mobile device by scanning the QR code affixed to the pump.

At the time of scanning, the code is associated with a generic link (linkID = “abcd”) that does not include parameters that specifically identify the pump, as indicated by the blanks in the database. Upon scanning, the payment application accesses a back-end processing system, e.g. hosted by the payment application provider, to initiate payment. At this time, it is determined that the link associated with the code does not encode merchant and location specific parameters, as seen by the empty database row. The user is then requested, via a user interface (shown as chat bubbles in Fig. 1) to provide the required information. Upon receipt of the information, the correct pump at the vendor location is activated, and the payment is processed and credited to the right vendor.

Further, the user-provided information is utilized to update the database, as shown on the right-hand size of Fig. 1. The information in this case includes the merchant (BestGas), location (City X, Street Y), and physical asset (Pump 1). Upon update of the database, subsequent scans of the code automatically identify the specific pump, as seen in the chat bubble in the right-hand side of Fig. 1.

In this manner, a large number of codes (stickers) that each encode a respective generic link can be utilized at a time of deployment. This is tolerant of human error, since any code can be affixed to any physical location, rather than specifically identifying a specific code for each location. For example, the payment application provider can generate a large number of merchant-specific stickers (e.g., for BestGas) that can be used at any location of the merchant. During field use, the associated link is automatically updated, thus reducing scope for errors in payment processing. After the database update, each code is uniquely linked to the corresponding physical asset.

As seen in the illustrative example of Fig. 1, the gas station can affix codes to any pump without worrying about error, and during operation, as users attempt payments via the codes, the payment application requests the user for information that specifically identifies each asset. If the user permits, the payment application can utilize data such as device location from the user device. With continued use of each pump, obtained user-generated data from multiple instances can be utilized to refine the generic link into an extended/refined link that includes the specific location and pump identifier. Reliability of the refinement process can be ensured by requiring a threshold number of matching user-initiated transactions for each code prior to updating the code.

In some implementations, pre-populated options may be provided to a user for confirmation/editing. This can also enable subsequent correction of the linked data, for cases where the stickers may be moved from one asset to another.

GENERIC BASE LINKS

```
http://paymentlink/gasdiscovery?_refinerid=${id1}
...
http://paymentlink/gasdiscovery?_refinerid=${idN}
```

USER PROVIDED DATA (FOR REFINEMENT)

```
{
base_url: "gaspayment"
params: {store_id: 1; pump: 1}
}
```

AFTER REFINEMENT

```
http://paymentlink/gaspayment?pump=1&store_id=1
```

Fig. 2: Embedded links updated (refined) with specific details based on user provided data

Fig. 2 depicts an example of refinement of an embedded link, per techniques of disclosure. In this illustrative example, an enterprise associated with gasoline distribution deploys a touchless code-based payment solution to a large number of gas pumps over multiple locations. Prior to deployment of the codes, a large batch of generic base links are generated, each associated with a unique identifier and are provided to the merchant for distribution across their sites. After refinement, the link associated with each code is updated to reflect the individual physical asset that the code is associated with.

The described link refinement techniques can be utilized in any setting that involves use of codes that are associated with specific physical assets, e.g. food ordering, movie ticketing, parking, etc. Further, the link refinement can be performed by any user. For example, an employee of the merchant can initiate payments via each physical asset and enter the associated asset information, prior to use by a customer, e.g., at a time of deploying the codes.

CONCLUSION

Payment applications make use of QR codes, NFC tags, etc. to encode information related to particular merchant physical assets and locations. Large scale deployment of such codes is difficult and prone to human error since the payment application provider needs to access merchant information to generate codes specific to each asset and the merchant needs to deploy the codes to specific matching locations. This disclosure describes the use of codes with unique but generic embedded links that can be deployed at any physical asset. When a payment is attempted by scanning such a code, the user is requested to provide information that serves to identify the physical asset. When a sufficient number of reliable user reports are received, the generic link associated with the code is updated to include the user-provided information to convert it into an asset-specific link that can be utilized to associate payments via the payment application with the specific physical asset.