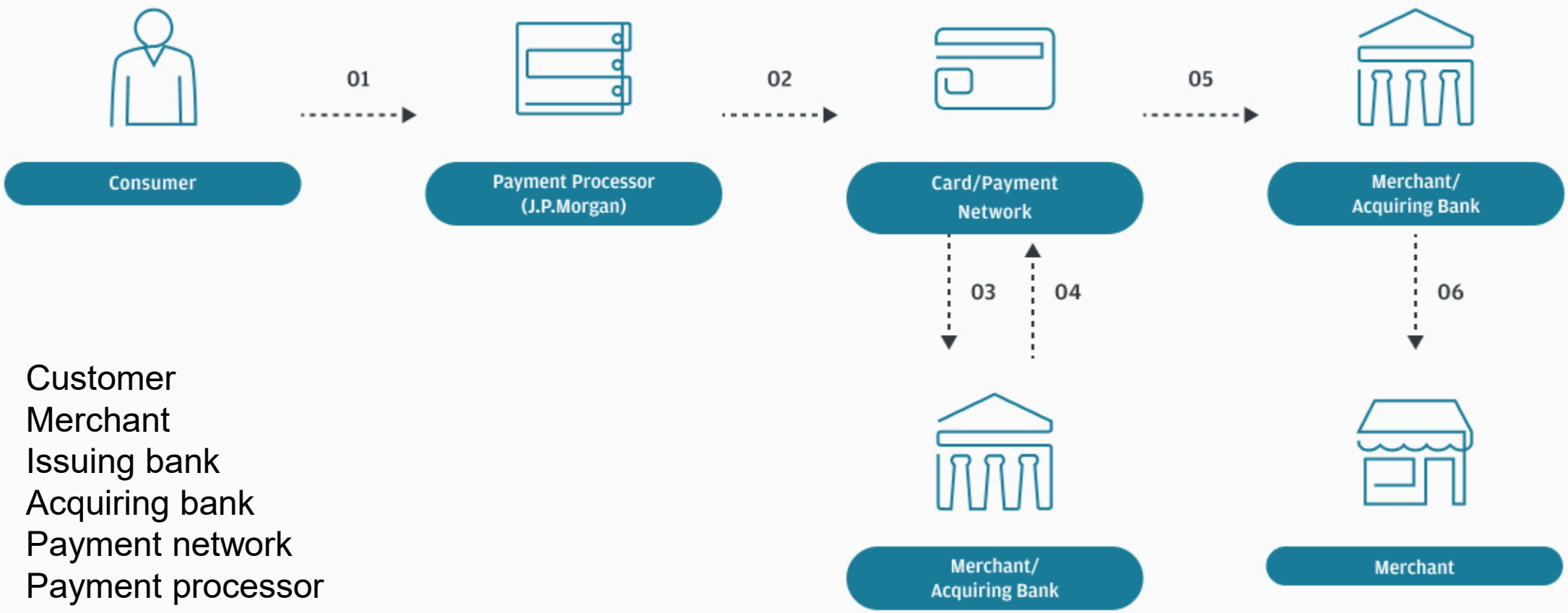
JPM Payments

Technical Book

JPM Products

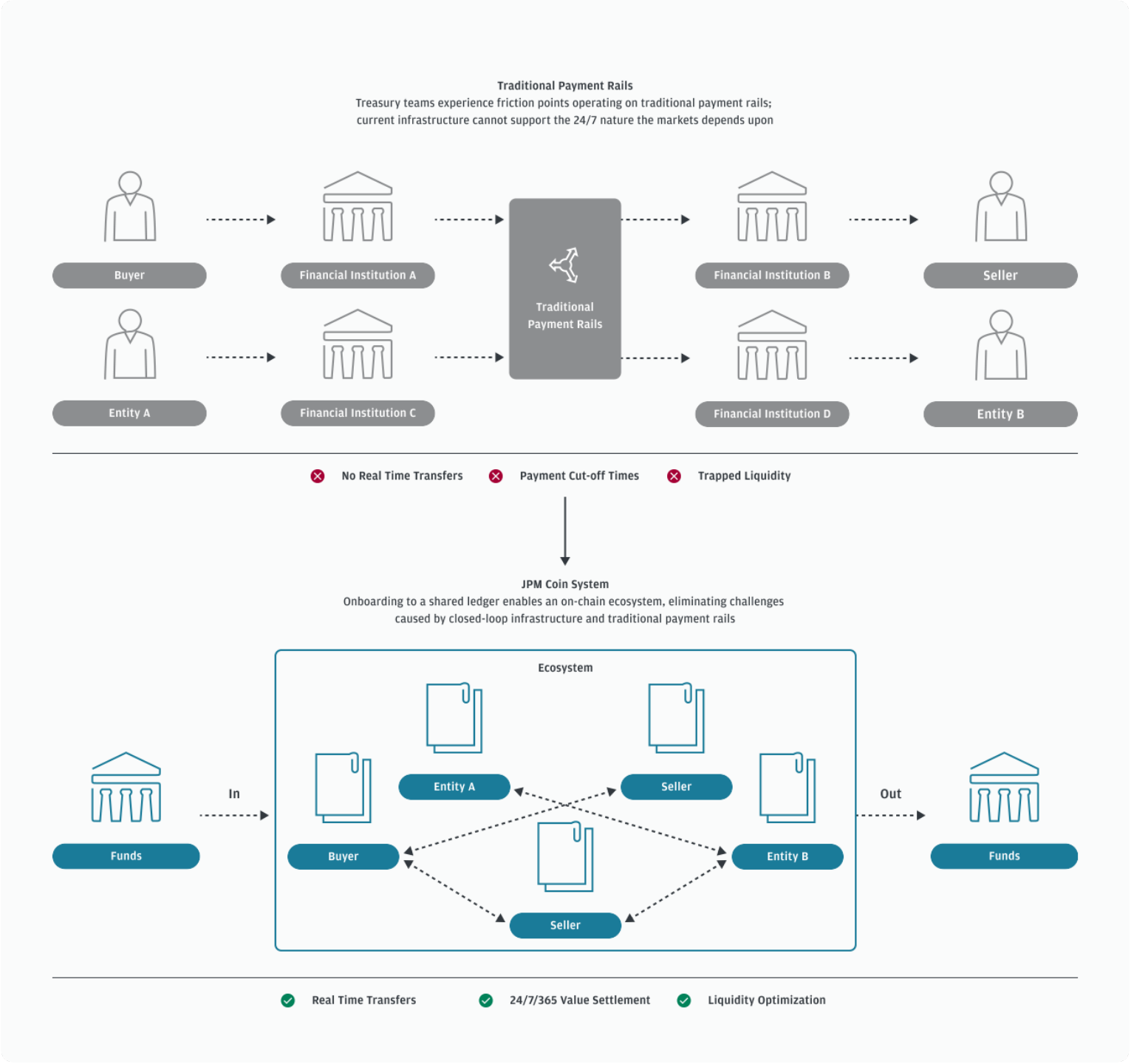
Real-time payments
Push to card
Automated clearing house
JPM Coin System

Payments lifecycle

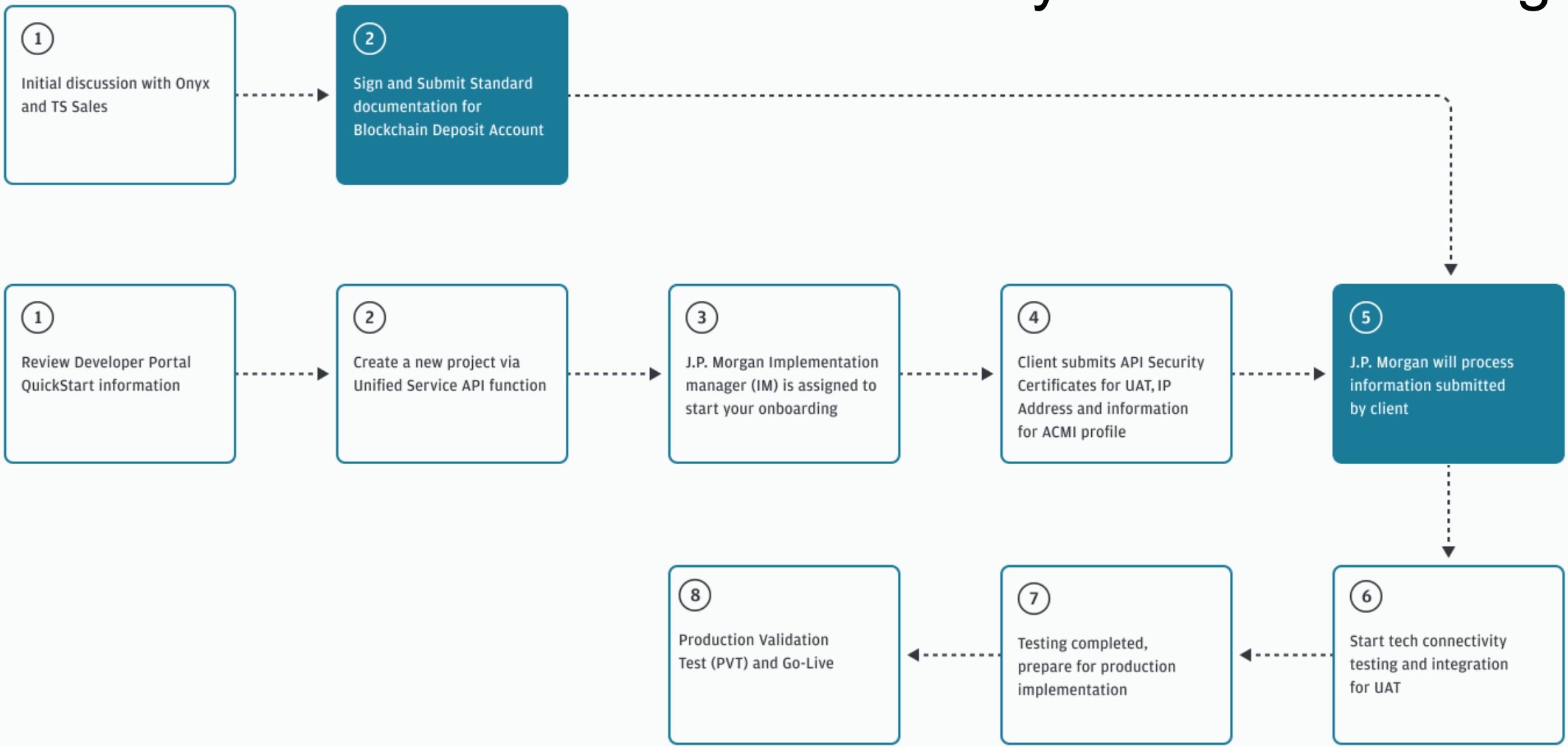


Customer
Merchant
Issuing bank
Acquiring bank
Payment network
Payment processor

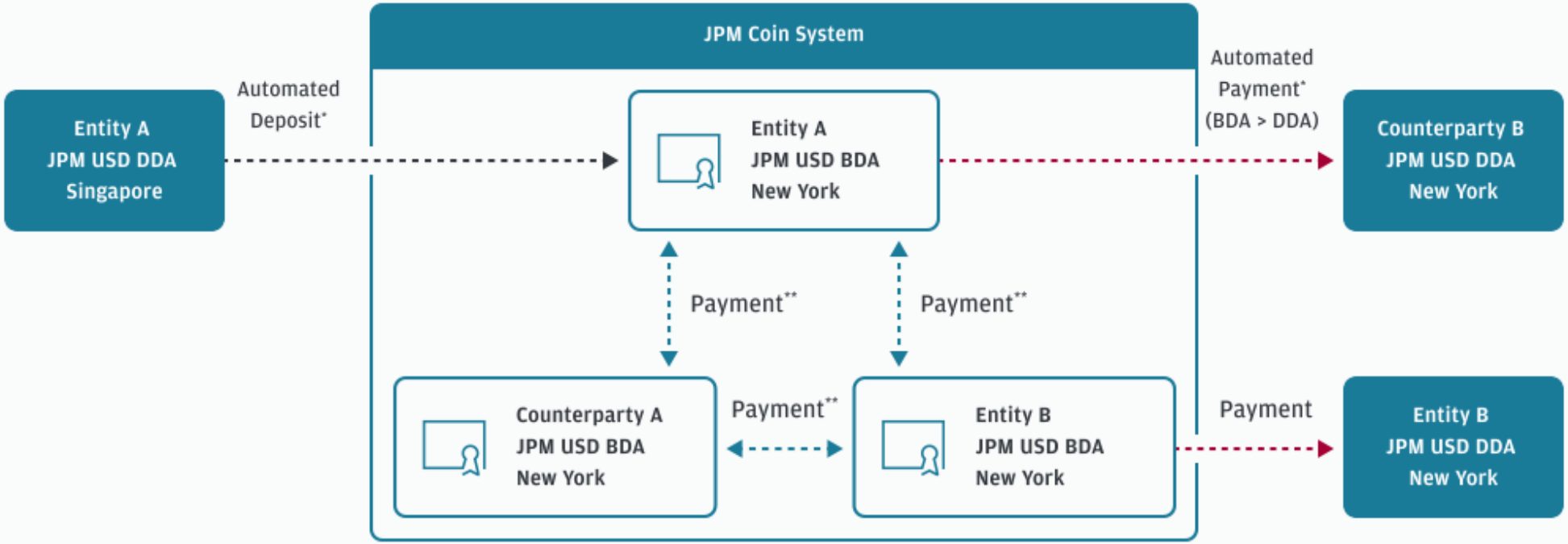
JPM Coin System



JPM Coin System Onboarding



JPM Coin System



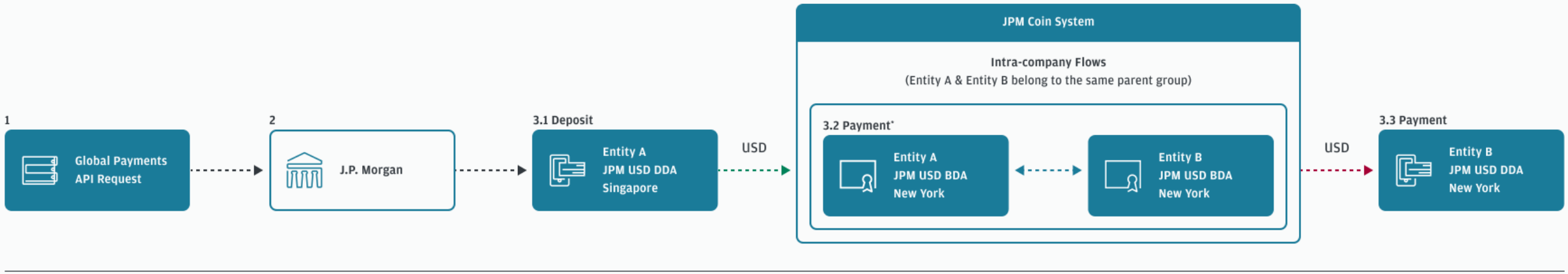
* The JPM Coin System offers a feature that allows the automated funding of a BDA and the withdrawal of funds from a linked BDA to DDA. Please reach out to your J.P. Morgan Payments Representative to discuss this feature.

** Payments on the JPM Coin System are completed on a 24/7/365 and on a same-day basis. Moving funds between systems - i.e., to and from the JPM Coin System and traditional DDA on legacy systems has a three-hour downtime over the weekend (3-6 PM EST every Saturday, enhancement under development)

Legend

- Blockchain Deposit Account (BDA)
- Payment (BDA to BDA)**
- Deposit (DDA to BDA)
- Payment (BDA to DDA)

Intra-company flow



Use case: Entity A is using JPM Coin System to manage their treasury liquidity during a holiday which requires moving funds from their USD Singapore DDA to their USD New York DDA

Payment Flow of Events

1. Client initiates payment via Global Payments API
2. J.P. Morgan validates JSON format and conducts mandatory data element checks
3. Instruction is sent to JPM Coin System for processing
   - 3.1 Entity A initiates a Deposit transaction to fund their BDA. Transaction will be completed, and client is sent payment status
   - 3.2 Entity A initiates a Payment transaction to move funds from Entity A's BDA to Entity B's BDA. Transaction will be completed, and client is sent payment status
   - 3.3 Entity B initiates a Payment transaction to move funds from Entity B's BDA to Entity B's DDA. Transaction will be completed, and client is sent payment status

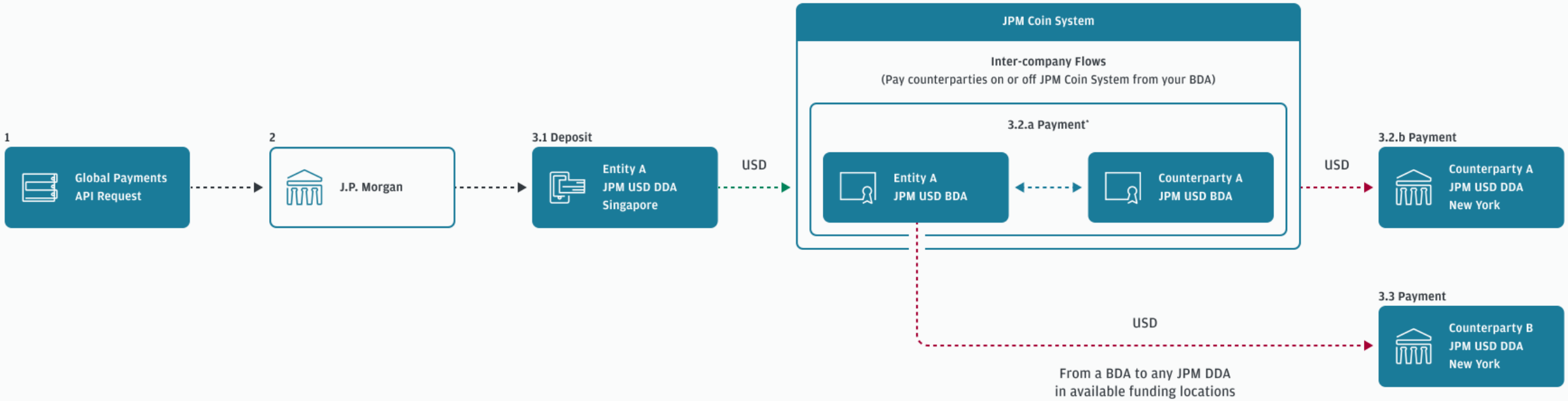
Legend

- Blockchain Deposit Account (BDA)
- Payment (BDA to BDA)*
- Deposit (DDA to BDA)
- Payment (BDA to DDA)

* Payments on the JPM Coin System are completed on a 24/7/365 and on a same-day basis. Moving funds between systems - i.e., to and from the JPM Coin System and traditional DDA on legacy systems has a three-hour downtime over the weekend (3-6 PM EST every Saturday, enhancement under development)

JPM Coin System

Inter-company flow



Use case: Entity A is using JPM Coin System to pay two different counterparties during the weekend from their USD Singapore DDA. Counterparty A is on the JPM Coin System Network and Counterparty B is not on the network but has a JPM DDA.

Payment Flow of Events

1. Client initiates payment via Global Payments API
2. J.P. Morgan validates JSON format and conducts mandatory data element checks
3. Instruction is sent to JPM Coin System for processing
   - 3.1 Entity A initiates a Deposit transaction to fund their BDA. Transaction will be completed, and client is sent payment status.
   - 3.2.a Entity A initiates a Payment transaction to pay counterparty A's BDA from Entity A's BDA. Transaction will be completed, and client is sent payment status
   - 3.2.b Counterparty A initiates a Payment transaction to move funds from their BDA to their DDA. Transaction will be completed, and client is sent payment status
   - 3.3 Entity A initiates a Payment transaction to pay directly from Entity A's BDA to Counterparty B's DDA. Transaction will be completed, and client is sent payment status

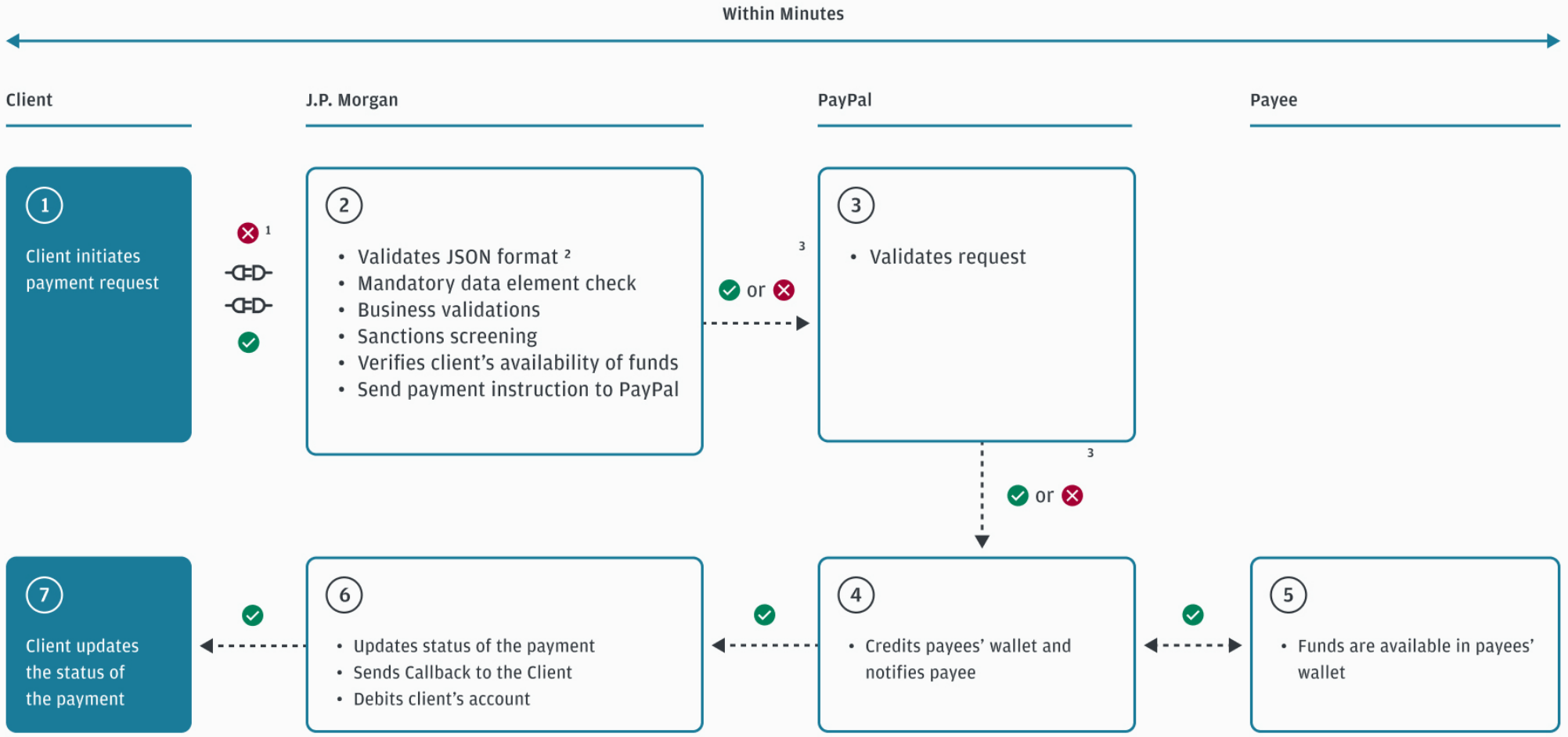
Legend

- Blockchain Deposit Account (BDA)
- Payment (BDA to BDA)*
- Deposit (DDA to BDA)
- Payment (BDA to DDA)

* Payments on the JPM Coin System are completed on a 24/7/365 and on a same-day basis. Moving funds between systems - i.e., to and from the JPM Coin System and traditional DDA on legacy systems has a three-hour downtime over the weekend (3-6 PM EST every Saturday, enhancement under development)

JPM Coin System

Push to *wallet*



Transaction goes to next step

Client receives an error message

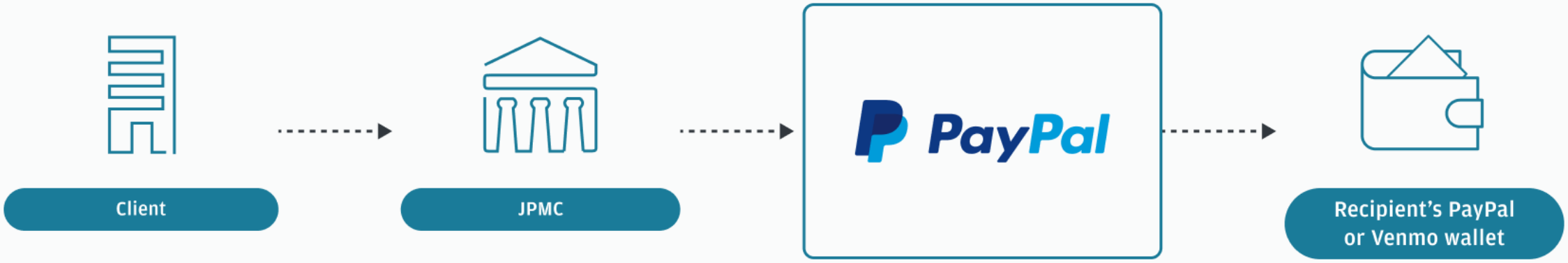
API or Webhook Callback

[1] Network error – Connection not made with API: Client should retry payment

[2] Connection made with API – Synchronous ACK or NACK is sent

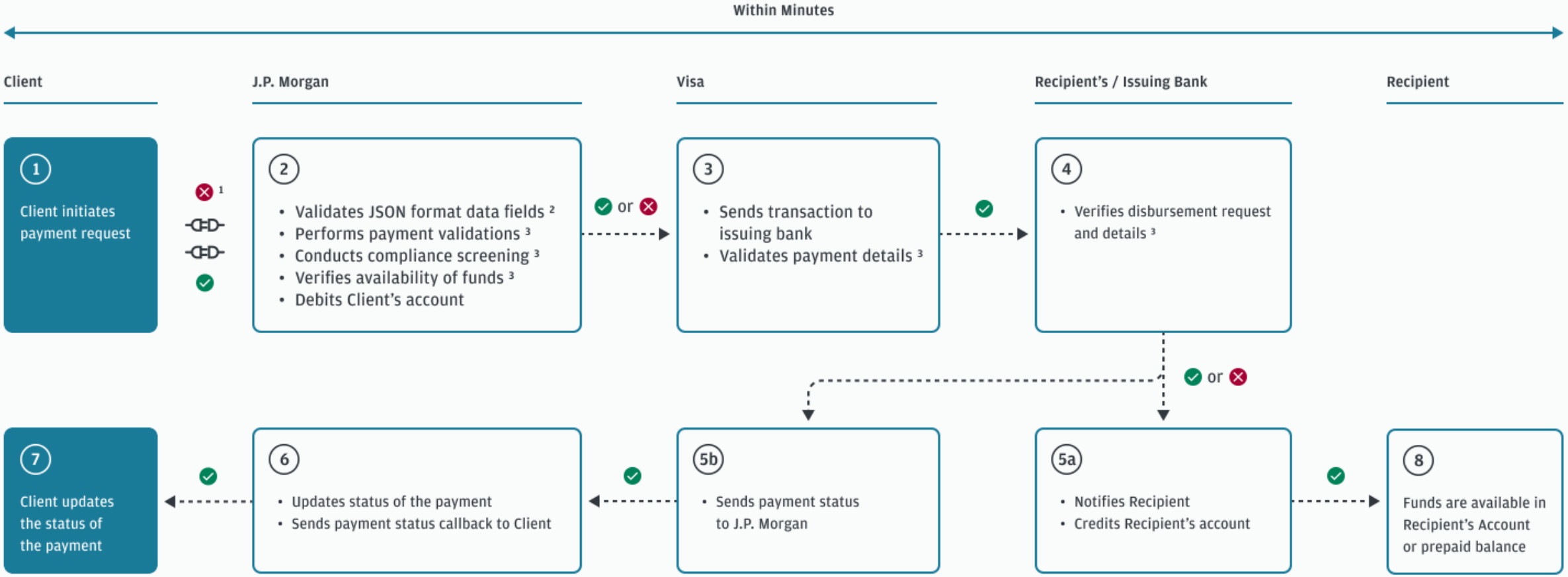
[3] If the result is a REJECT it will be sent via Callback with the appropriate error message

Benefits of push to *wallet*



- **Reach:** Access a network with millions of external wallet users, including recipients without bank accounts or who prefer alternate payment methods.
- **Speed:** Funds are quickly available to recipients.
- **Ease of use:** There is no need to capture bank account or routing numbers as you can use alias-based information, such as a mobile number or email address, to identify the recipient of the payment.

Push to card



Transaction goes to next step

Client receives an error message

API or Webhook Callback

[1] Network error – Connection not made with API: Client should retry payment

[2] Connection made with API – Synchronous ACK or NACK is sent

[3] If the result is a REJECT it will be sent via Callback with the appropriate error message

Push to card

BENEFITS

Coverage: This form of payment offers a nearly-ubiquitous footprint for consumers with a Visa or Mastercard-branded debit or reloadable prepaid card.

Ease of Use: There is no need to search for bank account or routing numbers as you can use the 16-digit debit card number and expiration date from your digital or physical wallet.

Speed: Funds are available in near-real time (30 minutes network SLA, but typically received in about 30 seconds).

JPM differentiator: Clients can fund and reconcile from the same account(s) used for other payment types.

When to use Push to Wallet

Insurance

Push to Wallet is a fast, easy method for insurers that want to offer consumers a seamless way to receive claim payouts.

Gig economy services

Push to Wallet can be used to pay out wages for short-term engagements, temporary contracts and independent contracting.

Telecom companies

Push to Wallet is a fast, easy method for telecom companies to pay out loyalty payments, promotions, refunds or carrier-switching rebates.

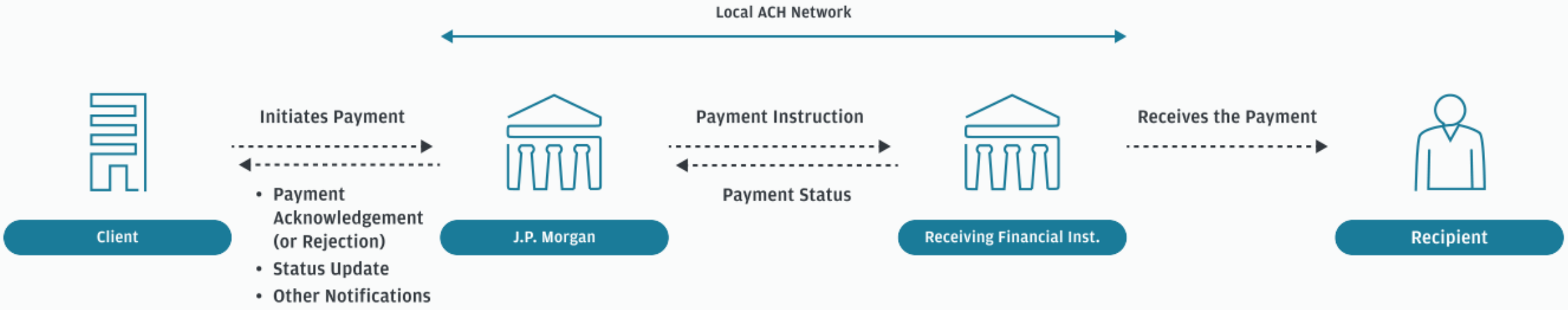
Rebates

Easily and quickly initiate rebates to customers.

Airlines

Make payments to recipients for lost baggage, employee compensation and more.

ACH flow


Local ACH Network
Initiates Payment
Payment Instruction
Receives the Payment
Payment Status
• Payment Acknowledgement (or Rejection)
• Status Update
• Other Notifications
Client
J.P. Morgan
Receiving Financial Inst.
Recipient

When to use ACH

Payroll payments

Pay employees their salary, wages, reimbursements and benefit payments directly into their bank account in a secure and cost effective way.

Supplier and vendor payments

Easily pay suppliers and vendors online, avoiding processing and later settlement associated with checks and corporate cards.

Bill and tax payments

Households can easily pay bills – for utilities and credit cards – and taxes directly from their bank account, and even set up a recurring debit arrangement with the biller for automatic bill payment.

Customer reimbursements and refunds

Issue refunds or reimbursements that reach customers in a few days.

Wallet top-up and withdrawal

Top-up stored-value digital wallets directly from bank accounts and withdraw excess money directly into bank accounts.

When to use Push to Card

Insurance claims

Push to Card is a fast, easy method for insurers that want to offer consumers a seamless way to receive claim payments.

Gig economy services

Ride share companies can use Push to Card as point of differentiation to attract and retain workers.

Telecom companies

Push to Card is a fast, easy method for telecoms that want to offer consumers financial incentives to switch from a competitor.

Government disbursements

Merchants can use Push to Card to disburse government payments including social security payments, unemployment, disability, and more.

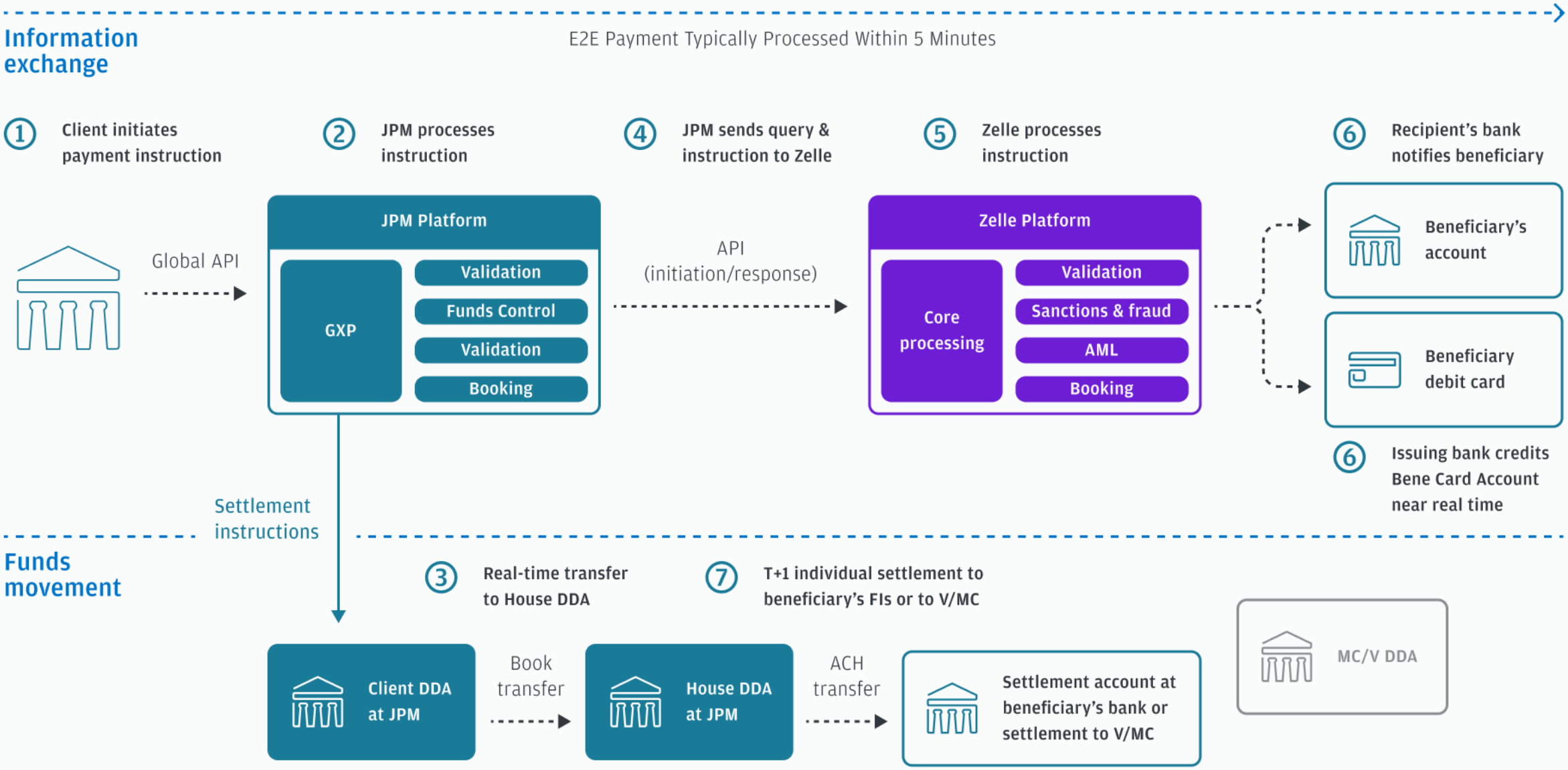
Marketplace payouts

Marketplaces can use Push to Card as an easy method for sellers to receive funds.

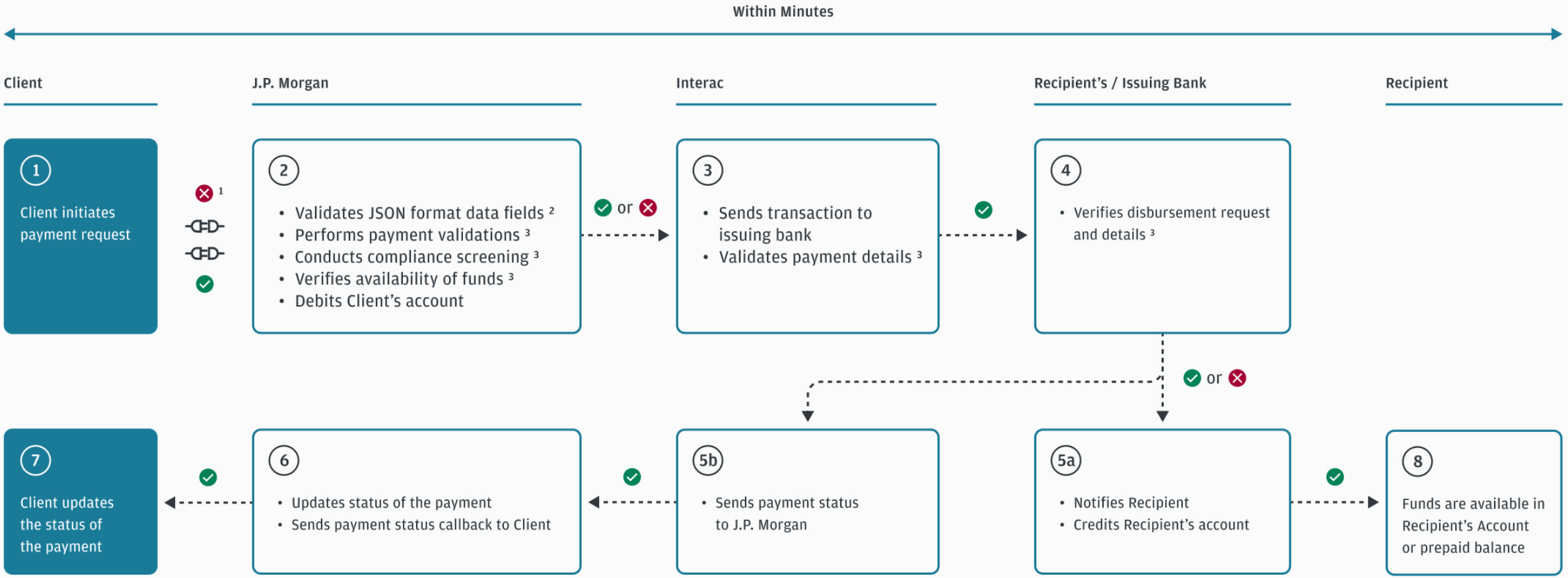
Airline payments

Make payments to recipients for lost baggage, employee compensation and more.

Zelle use case



Intertac payments use case



Transaction goes to next step

Client receives an error message

API or Webhook Callback

[1] Network error – Connection not made with API: Client should retry payment

[2] Connection made with API – Synchronous ACK or NACK is sent

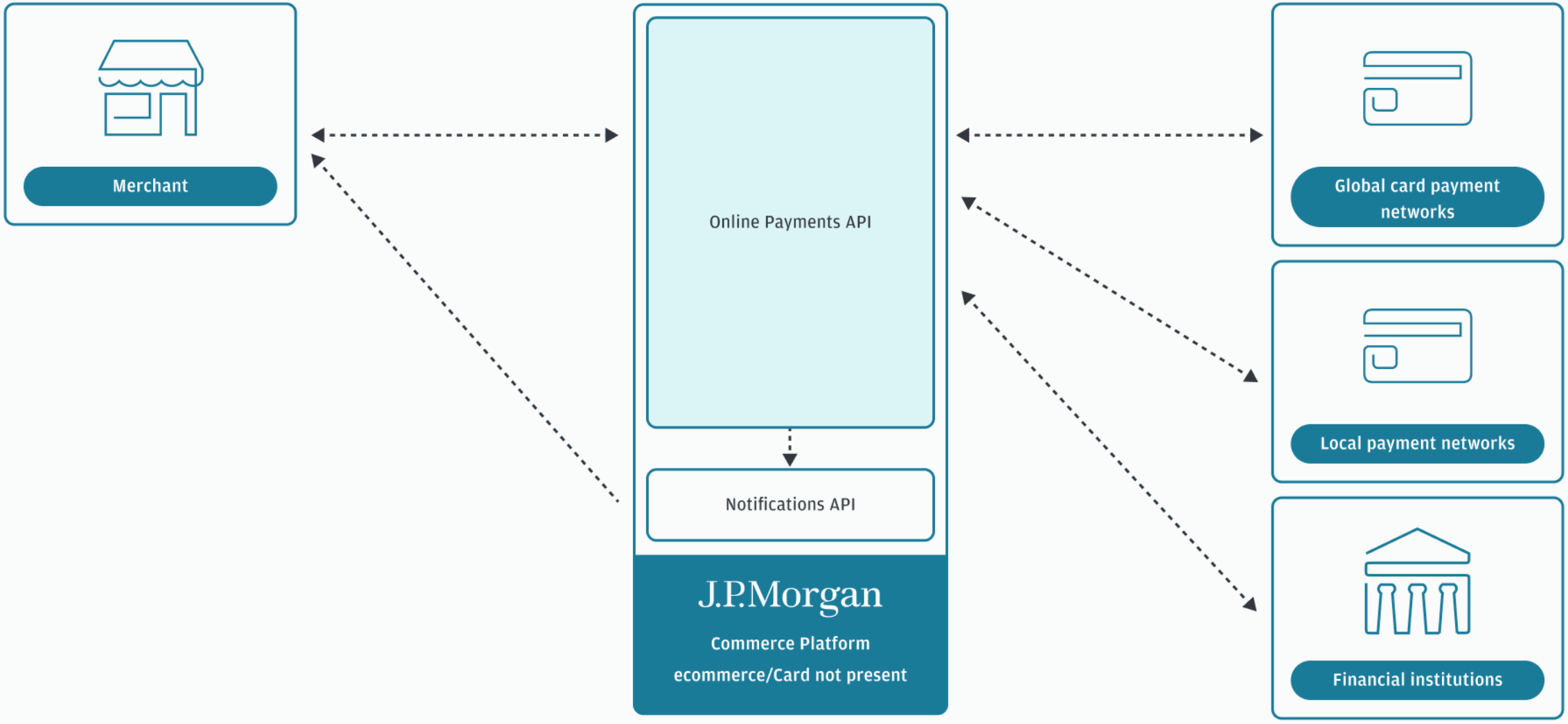
[3] If the result is a REJECT it will be sent via Callback with the appropriate error message


Sender
J.P. Morgan
Interac
Beneficiary bank
Beneficiary

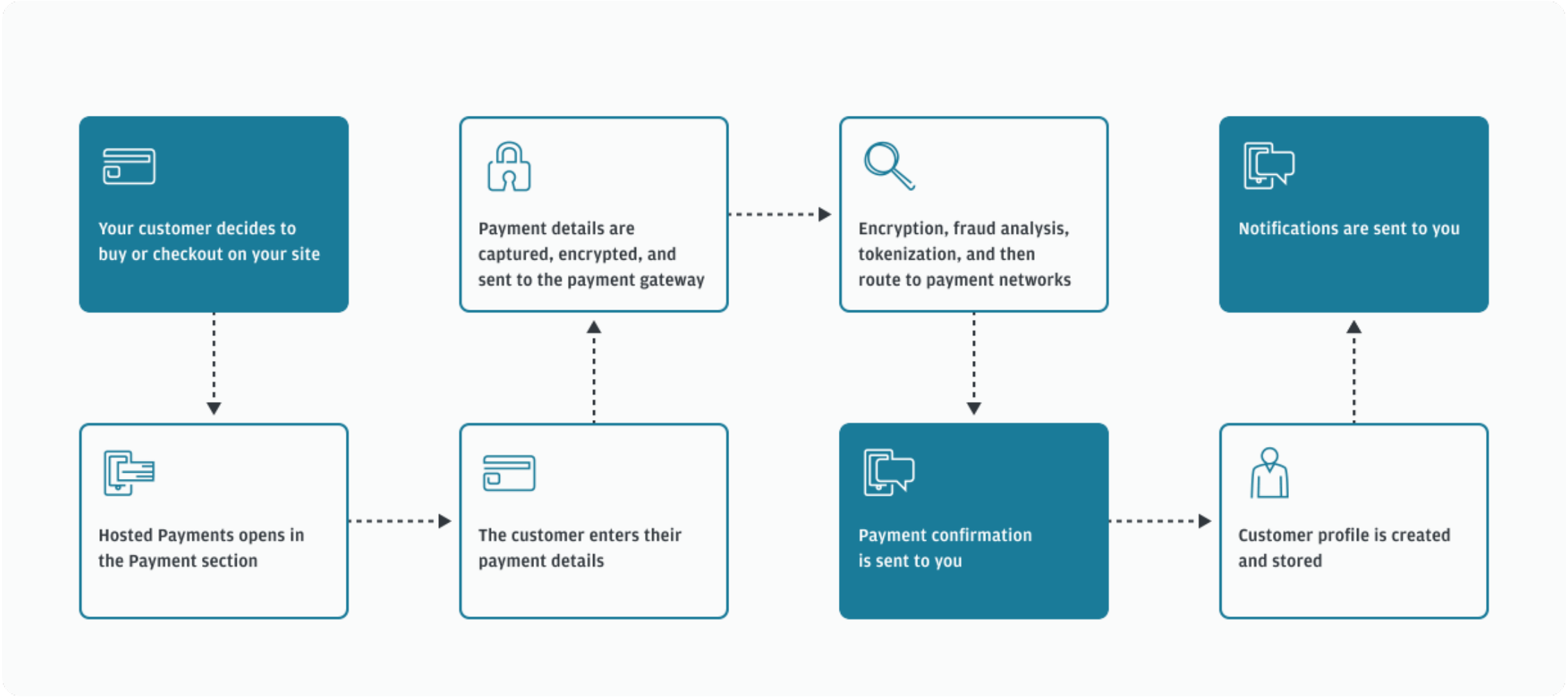
Online Payments

Accept and manage online payments with our direct Online Payments API integration:

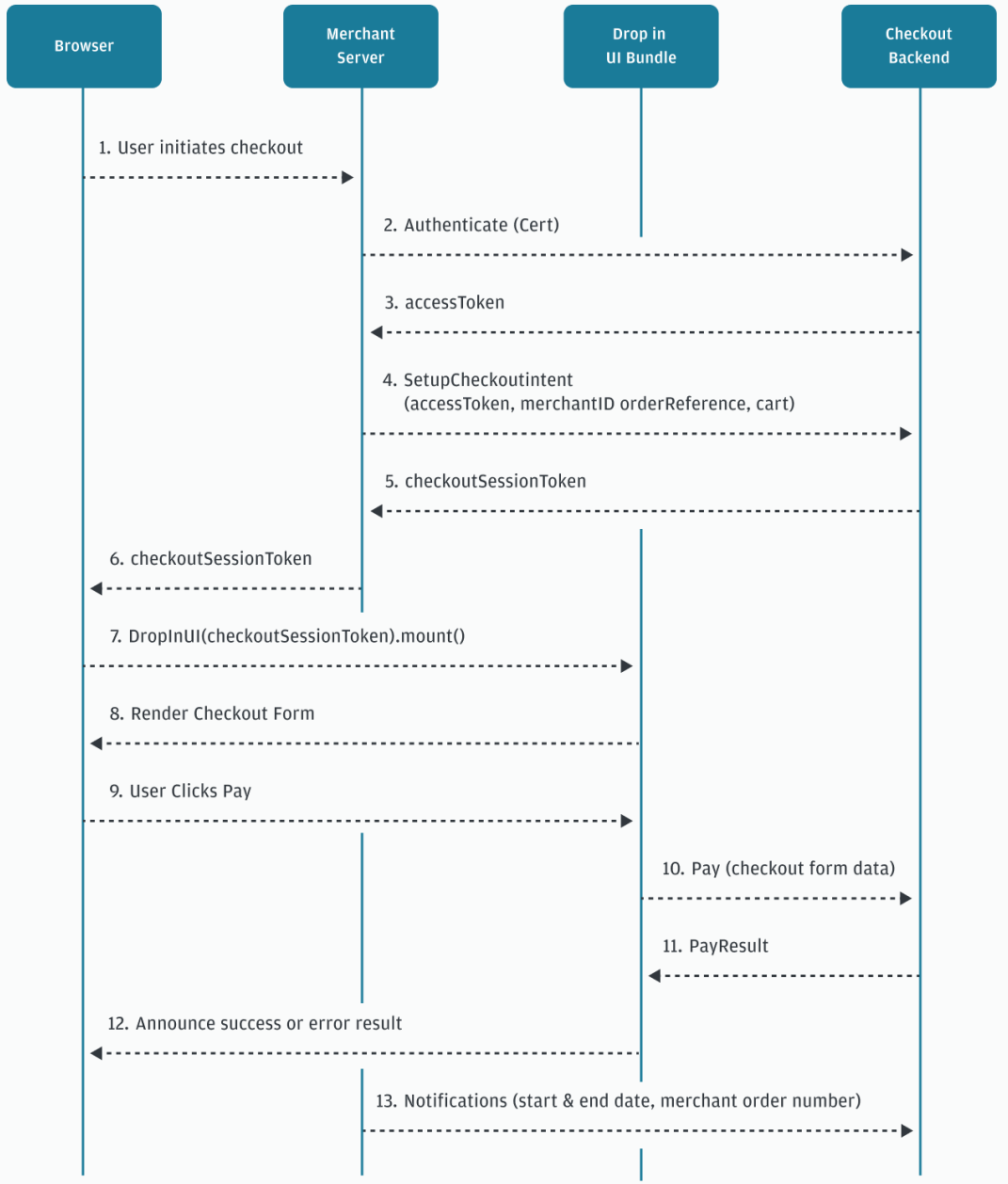
- Process card, wallet, local, and alternative methods of payment
- Store or set up recurring payments
- Protect payment information with encryption and fraud tools
- Easily manage consumer profiles


Merchant
Online Payments API
Notifications API
J.P.Morgan
Commerce Platform
ecommerce/Card not present
Global card payment networks
Local payment networks
Financial institutions

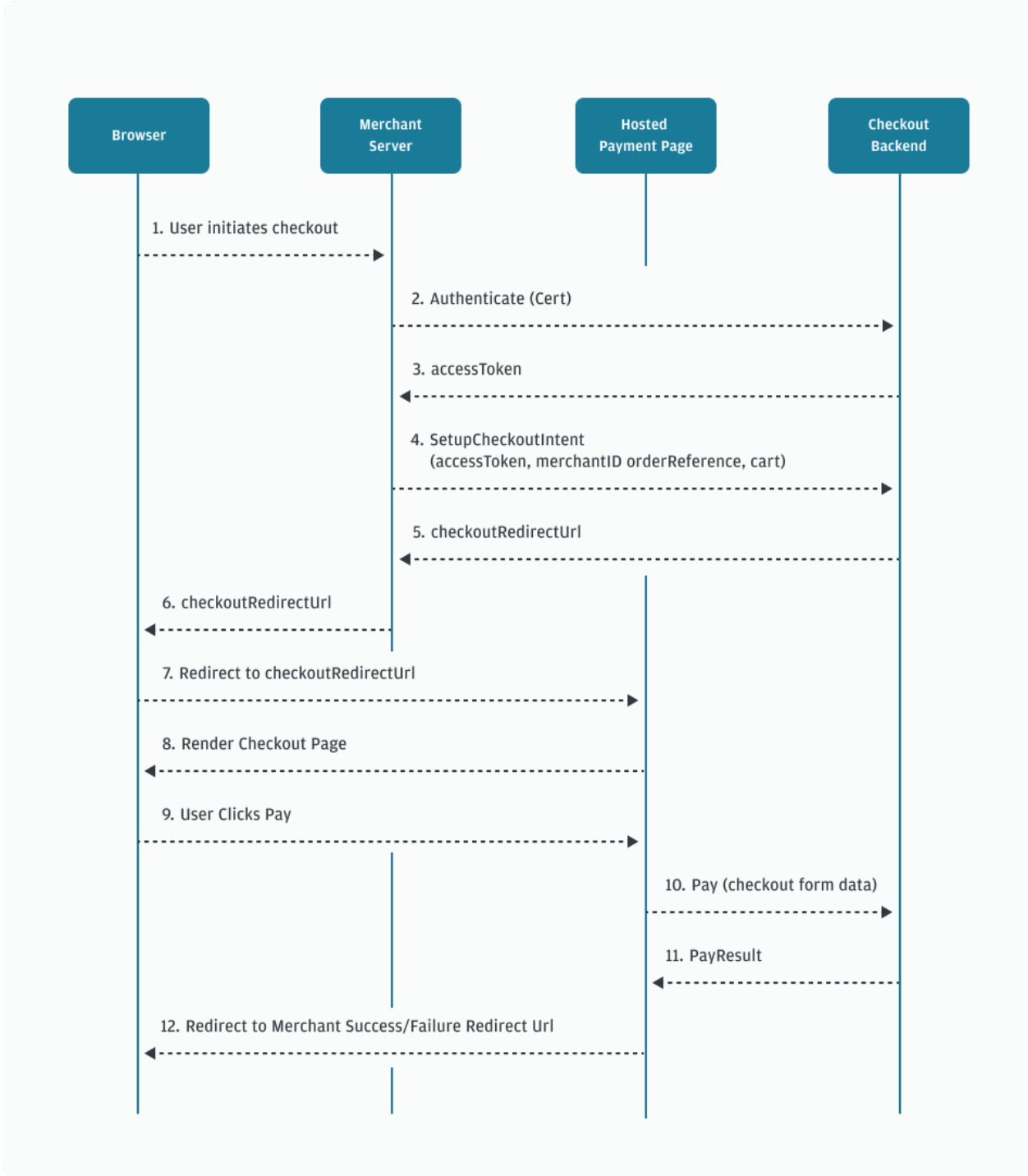
Checkout


Your customer decides to buy or checkout on your site
Hosted Payments opens in the Payment section
The customer enters their payment details
Payment details are captured, encrypted, and sent to the payment gateway
Encryption, fraud analysis, tokenization, and then route to payment networks
Payment confirmation is sent to you
Customer profile is created and stored
Notifications are sent to you

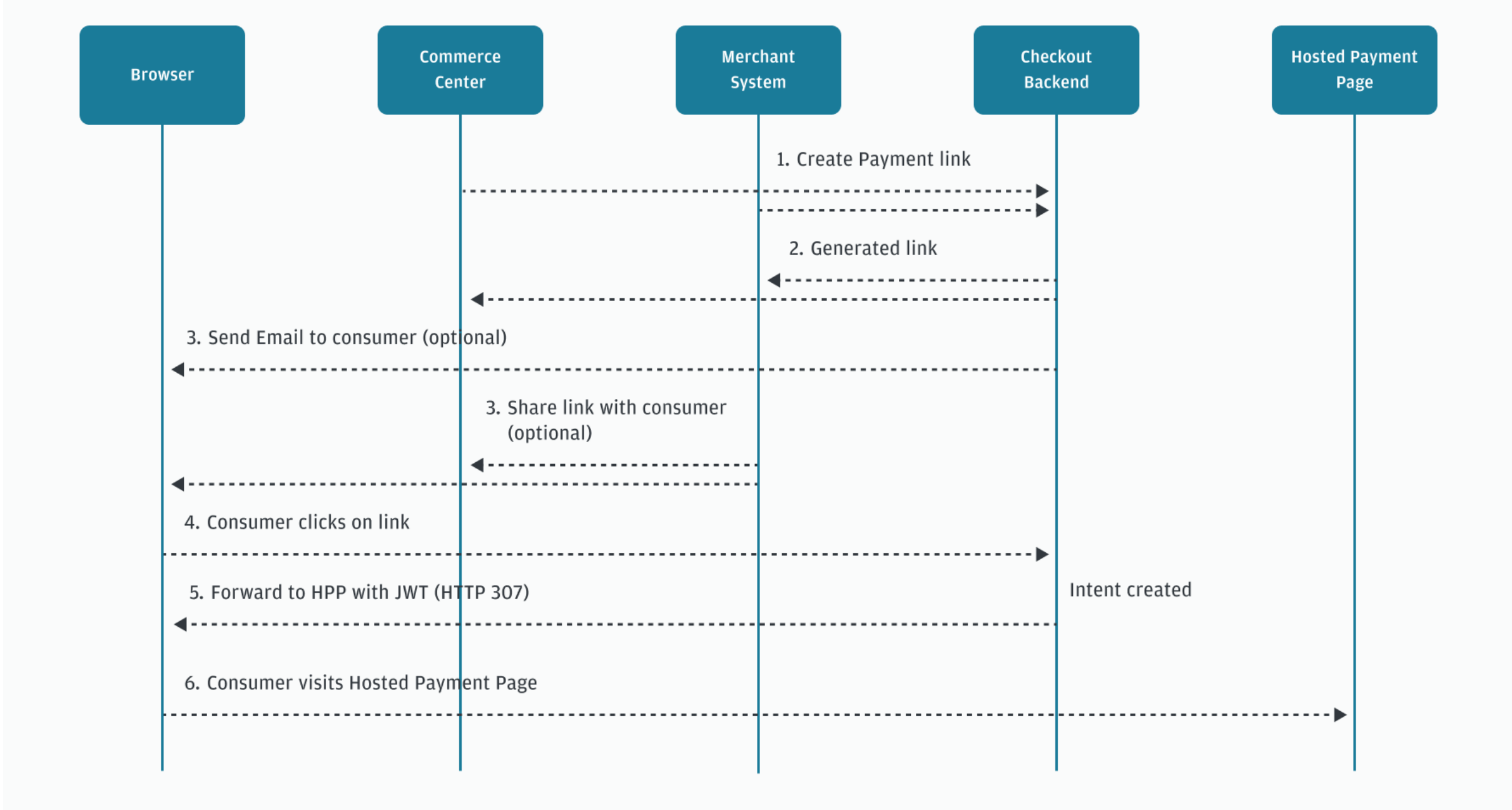
Drop-in UI


Browser
Merchant Server
Drop in UI Bundle
Checkout Backend
1. User initiates checkout
2. Authenticate (Cert)
3. accessToken
4. SetupCheckoutintent
(accessToken, merchantID orderReference, cart)
5. checkoutSessionToken
6. checkoutSessionToken
7. DropInUI(checkoutSessionToken).mount()
8. Render Checkout Form
9. User Clicks Pay
10. Pay (checkout form data)
11. PayResult
12. Announce success or error result
13. Notifications (start & end date, merchant order number)

Hosted payments page

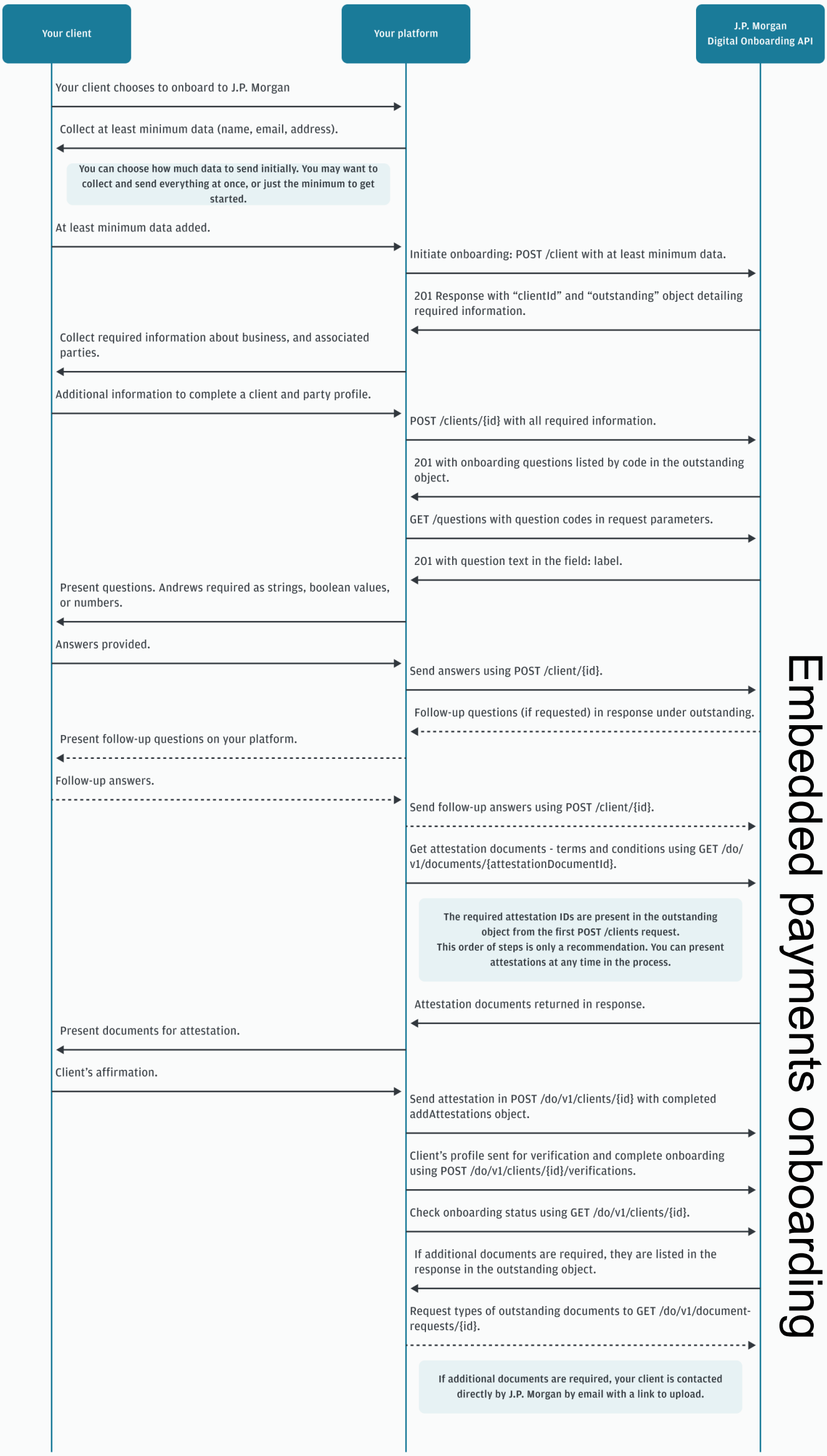


Pay by link


Browser
Commerce Center
Merchant System
Checkout Backend
Hosted Payment Page
1. Create Payment link
2. Generated link
3. Send Email to consumer (optional)
3. Share link with consumer (optional)
4. Consumer clicks on link
5. Forward to HPP with JWT (HTTP 307)
Intent created
6. Consumer visits Hosted Payment Page

Embedded payments onboarding

1. **POST /do/v1/clients** - send basic client information, create related parties to represent the client's business, controllers, and beneficial owners. In the response, you receive a clientId and partyIds. In addition, the response may contain outstanding onboarding requirements:
   - Outstanding party details
   - Due diligence question IDs.
   - Required attestations.
   - Required documents.
2. **GET /do/v1/questions** - to retrieve additional client onboarding questions that your client must answer. The IDs for these questions are returned in the POST /do/v1/clients response.
3. **POST /do/v1/clients/{id}** - to send more detailed information required for due diligence, including answers to outstanding questions.
4. **GET /do/v1/documents/{id}/file** - get documents that must be read and attested by your client.
5. **POST /do/v1/clients/{id}/verifications** - send an instruction to J.P. Morgan to perform the due diligence and verification checks to complete onboarding.
6. **GET /do/v1/clients/{id}** - to check the onboarding status of your client.


Your client
Your platform
J.P. Morgan Digital Onboarding API
Your client chooses to onboard to J.P. Morgan
Collect at least minimum data (name, email, address).
You can choose how much data to send initially. You may want to collect and send everything at once, or just the minimum to get started.
At least minimum data added.
Initiate onboarding: POST /client with at least minimum data.
201 Response with “clientId” and “outstanding” object detailing required information.
Collect required information about business, and associated parties.
Additional information to complete a client and party profile.
POST /clients/{id} with all required information.
201 with onboarding questions listed by code in the outstanding object.
GET /questions with question codes in request parameters.
201 with question text in the field: label.
Present questions. Andrews required as strings, boolean values, or numbers.
Answers provided.
Send answers using POST /client/{id}.
Follow-up questions (if requested) in response under outstanding.
Present follow-up questions on your platform.
Follow-up answers.
Send follow-up answers using POST /client/{id}.
Get attestation documents - terms and conditions using GET /do/v1/documents/{attestationDocumentId}.
The required attestation IDs are present in the outstanding object from the first POST /clients request.
This order of steps is only a recommendation. You can present attestations at any time in the process.
Attestation documents returned in response.
Present documents for attestation.
Client’s affirmation.
Send attestation in POST /do/v1/clients/{id} with completed addAttestations object.
Client’s profile sent for verification and complete onboarding using POST /do/v1/clients/{id}/verifications.
Check onboarding status using GET /do/v1/clients/{id}.
If additional documents are required, they are listed in the response in the outstanding object.
Request types of outstanding documents to GET /do/v1/document-requests/{id}.
If additional documents are required, your client is contacted directly by J.P. Morgan by email with a link to upload.
Embedded payments onboarding

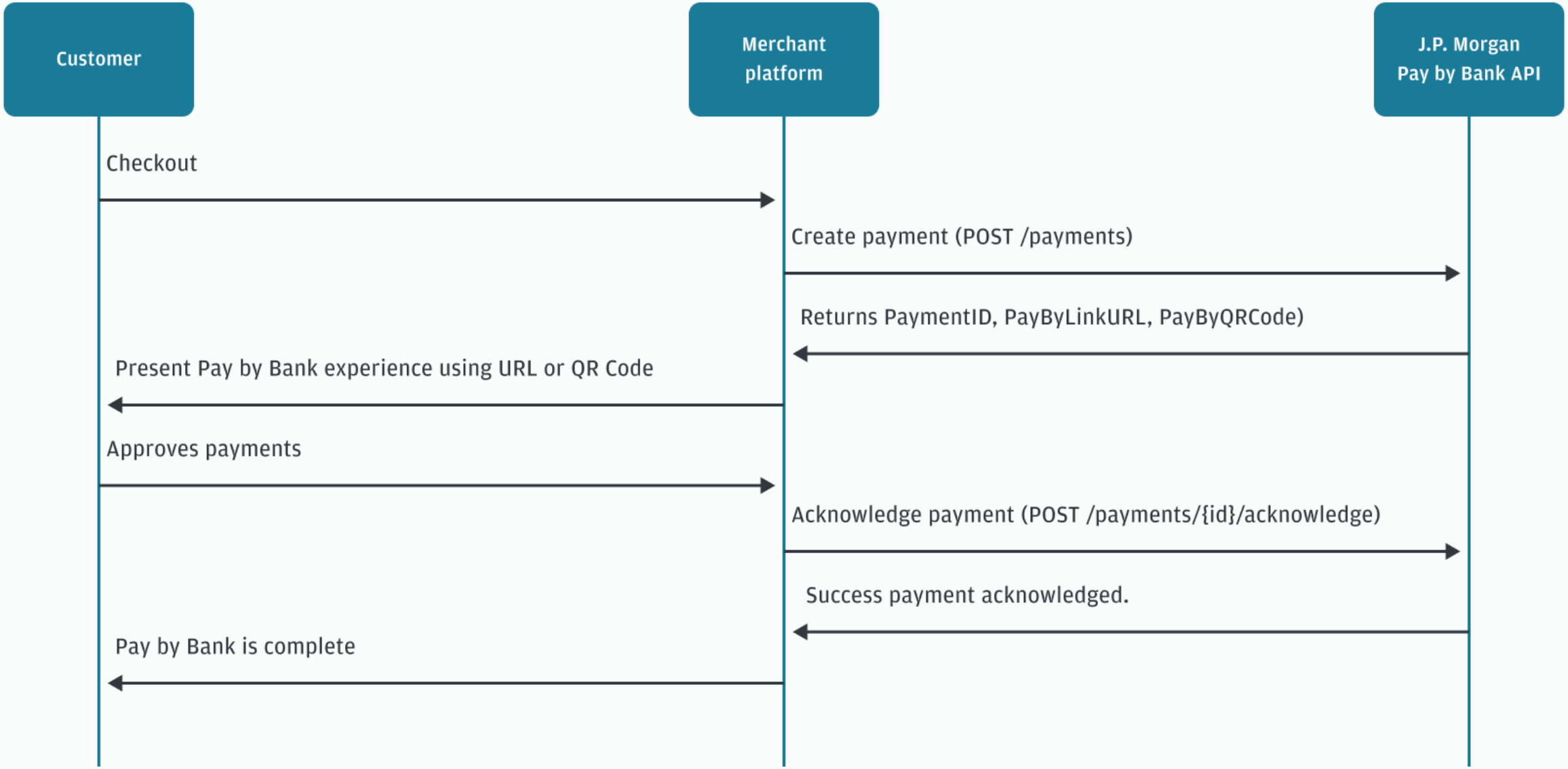
Initiate a payment

1. Your customer selects Pay by Bank as a payment option.
2. You initiate the payment using **POST /payments**.
3. The **POST** request returns an ID and a link which you display to your customer. This link can also be used to automatically open the Pay by Bank screens.
4. Your customer gives their permission through the website flow. There are no additional API calls to make during this process, the UX guides the customer through the payment process.
5. Once payment is complete, your customer is able to return to your checkout screens.
6. Using the ID returned in the original request, you send a request to **POST payments/{id}/acknowledge**
7. Display the payment acknowledgement details to your customer.
8. Retrieve payments with **GET /payments/{id}**

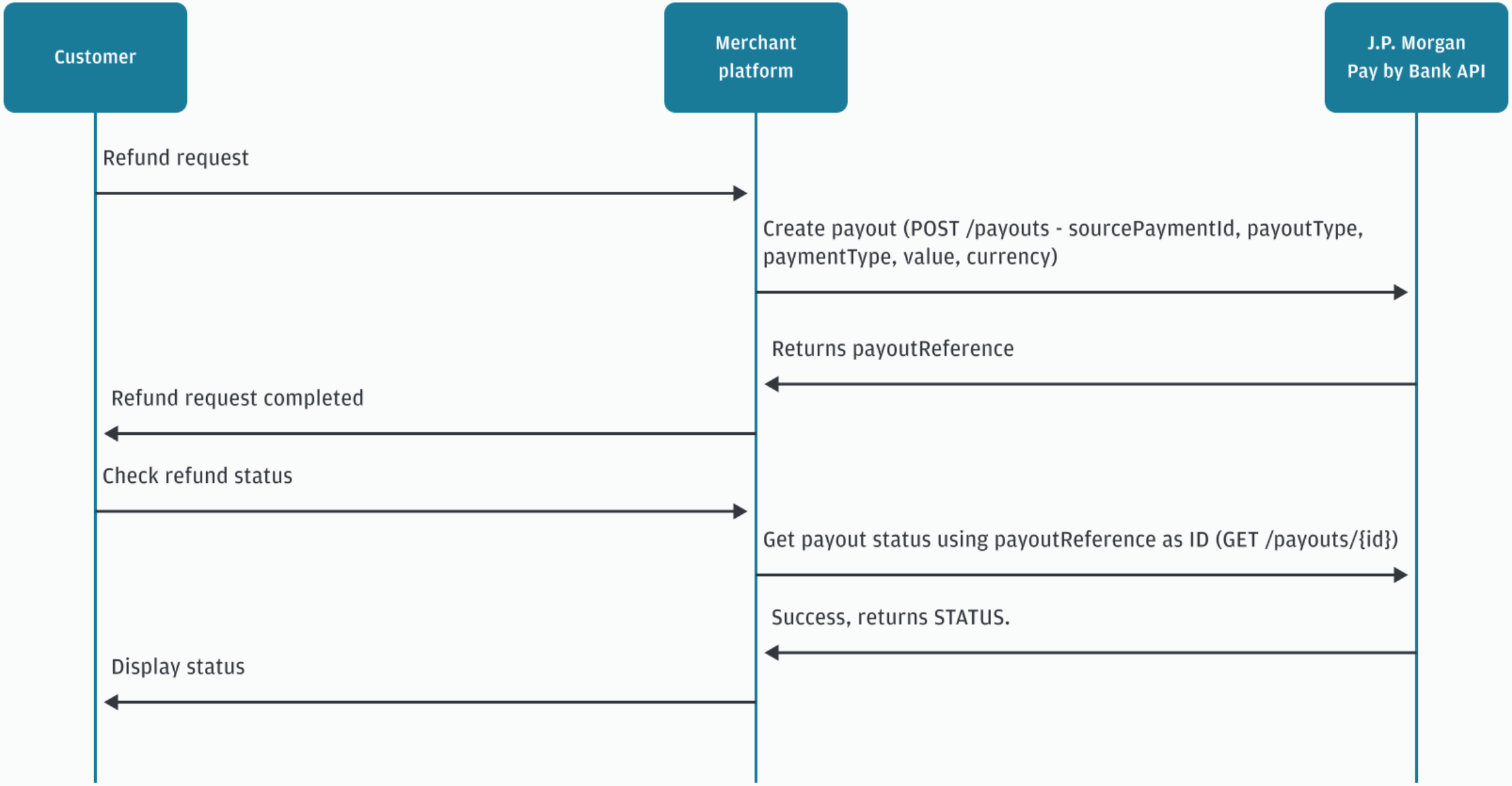
Needed parameters

debtor information - the name of the party making the payment.
market information - the ISO code for the market where the payment is being made.
creditor information - the party receiving the payment. Including name, account details and type.
reference - your unique ID for the payment.
paymentAmmount - the amount being paid and the currency.
paymentType - the type of payment being made. This can be RTP, LOW_VALUE_RETAIL or HIGH_VALUE_BUSINESS.

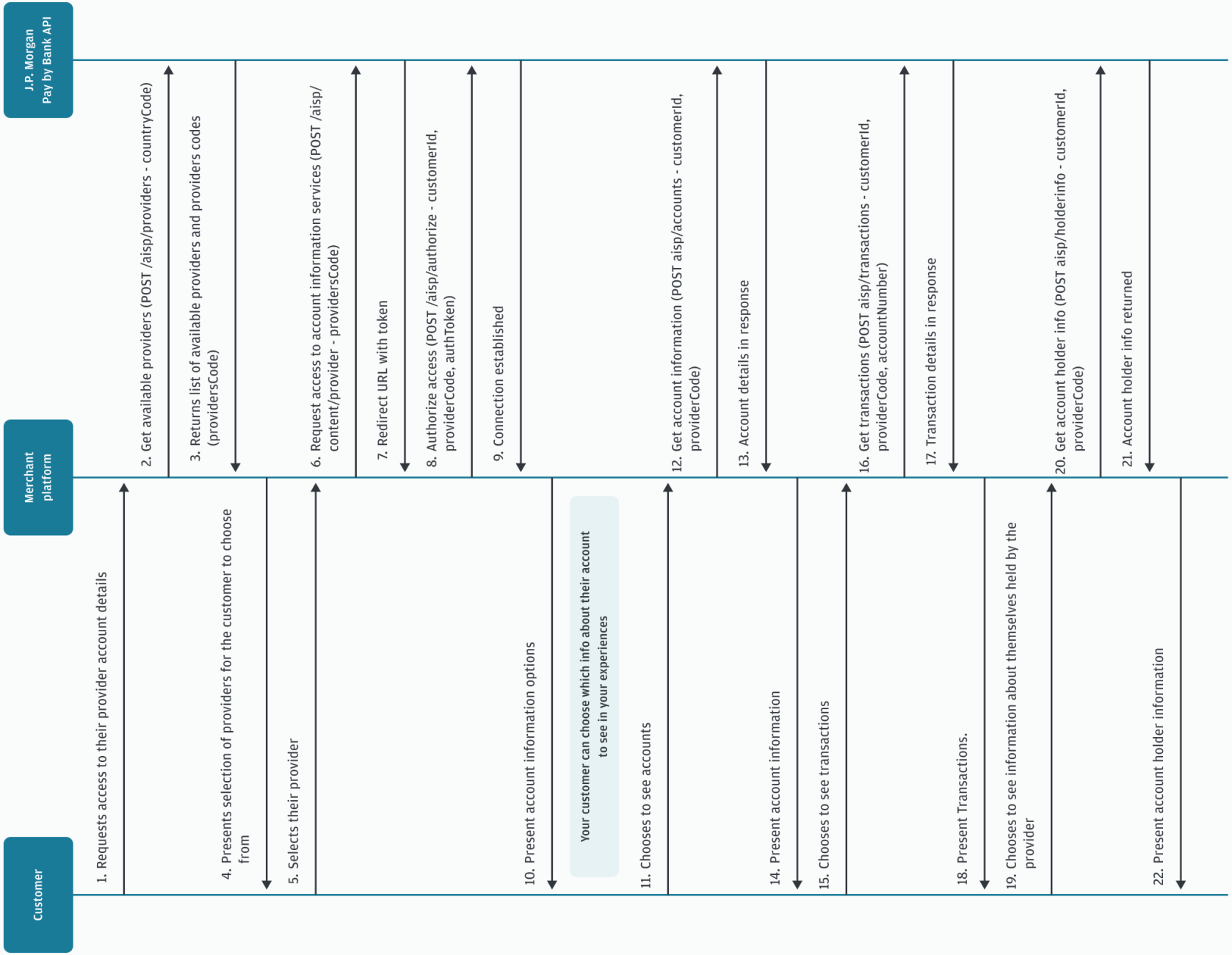
Payments initiation v2



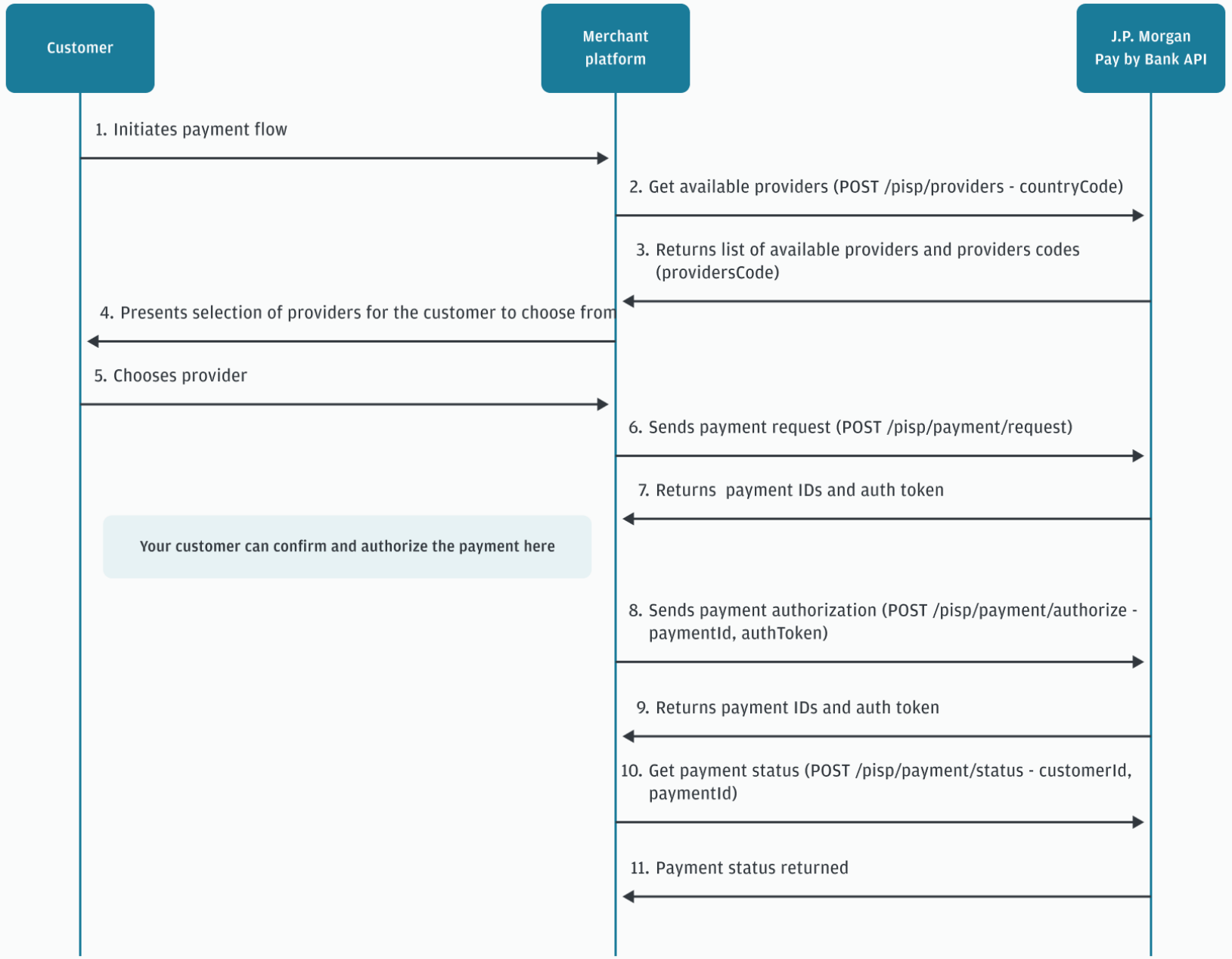
Request a refund v2 (payout)



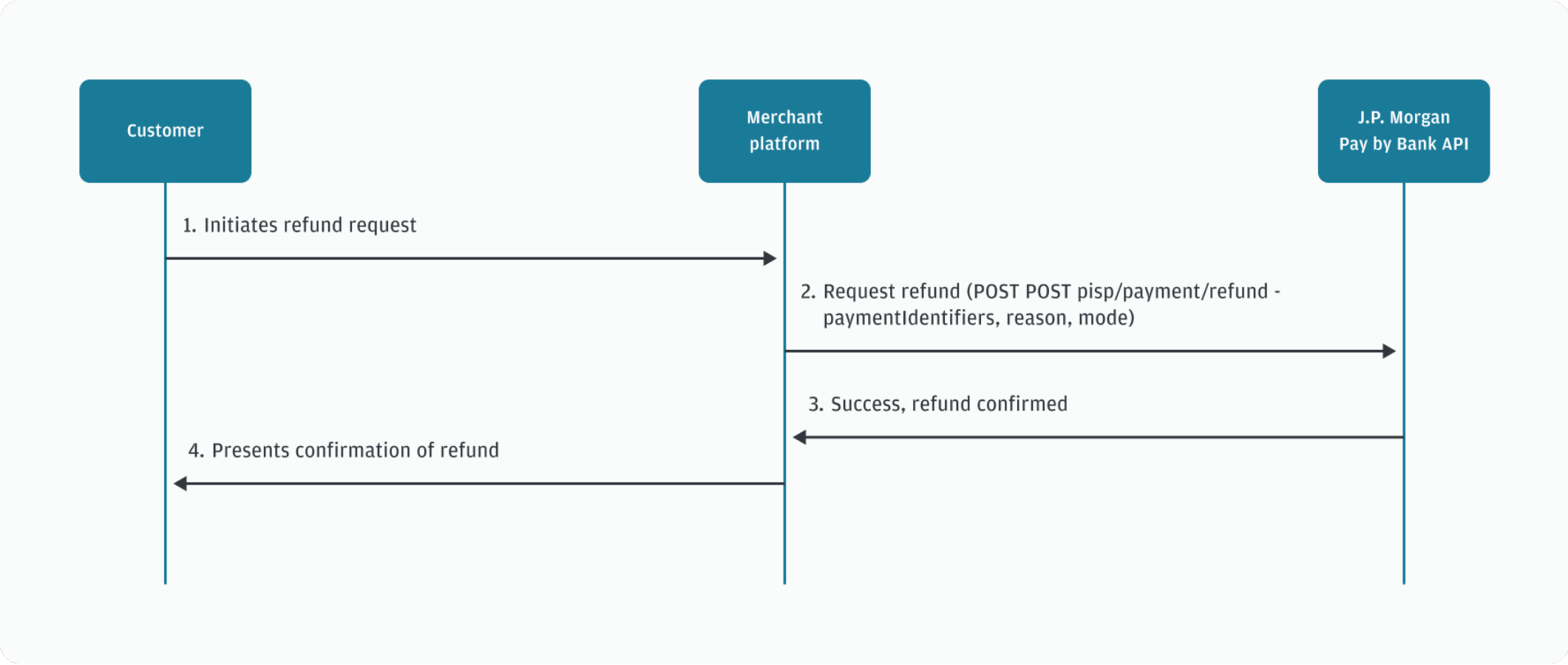
Account information service v1



Initiate a payment v1



Request a refund v1



Global payments API

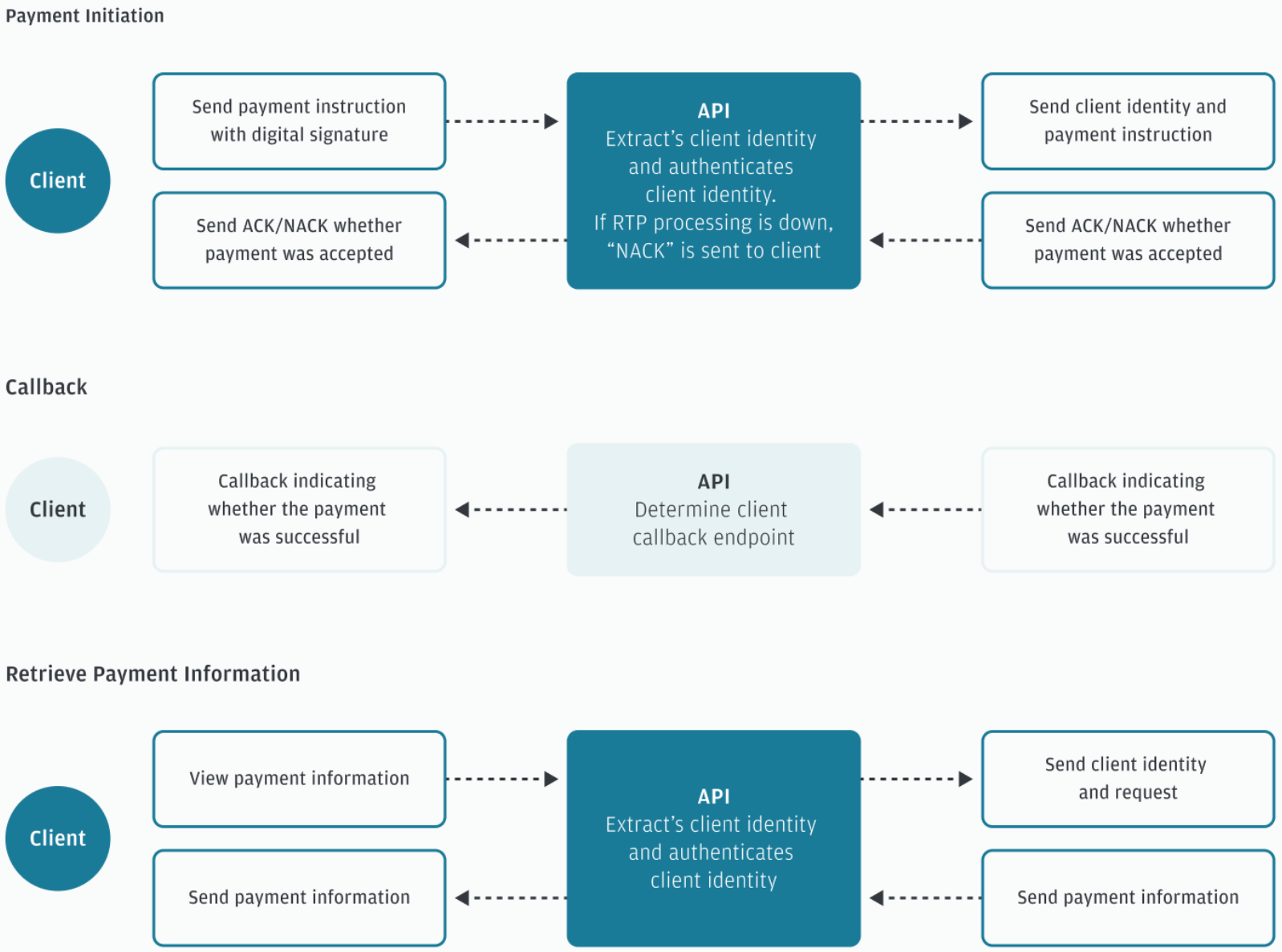
BENEFITS

1. Avoid prefunding with 24/7 real-time payments:	Move funds on banking holidays, fund shortfalls, working capital gaps, and other ad hoc transfers.
2. Cross-border payments:	Unlock working capital globally with cross-border payments and enable new business models, for example, machine-to-machine payments.
3. Real-time visibility:	View transactions and balances in real-time to ensure transparency and finality of payment.
4. Advance payment type support:	Use Delivery versus Payment (DvP), Payment versus Payment (PvP), and machine-to-machine payment types.

Send and track payments globally with a single API

- Near-instant payment processing
- Choose from over 5 payment methods
- Available daily in over 15 countries
- Track payment status to completion

Global payments



https://developer.payments.jpmorgan.com/