



## Boon Chapman Follow Up Items

### REST API Documents

- Account Updater
  - Overview - Account updater aims to close the gap on bad transaction responses due to card information updates resulting from account closures, card reported lost or stolen, expired cards, product upgrades and portfolio conversions between issuers. AU serves to relieve merchants of declines and helps maintain the integrity of the cardholder experience, reducing service interruptions, cancellations, and ultimately customer dissatisfaction
  - Benefits
    - Increases authorization approvals
    - Reduces opportunity for cancellation
    - Simplifies the CoF framework
    - Maintains integrity of support and may reduce service expenses
    - Strengthens customer relations
  - How It Works
    - On a monthly basis, Forte will present account records of all participating merchants' recurring transactions against the Account Updater database. Forte is notified of updates and modifies the token database with the new information (Account Number, Expiration date, Closed Account)
    - Merchant performs REST API 'GET' Account Updater Token Updates request of payment methods for updated tokens
    - REST API –
      - `{{baseURI}}/organizations/org_{{organizationID}}/paymethods/?filter=start_au_updated_date+eq+'2020-01-08'+and+end_au_updated_date+eq+'2020-12-08'`
      - Account Updater customers use this URI with the `start_au_updated_date` and `end_au_updated_date` filters to view the payment tokens that were updated in the prior months.
      - If the need to exclude specific credit cards from an Account Updater run was needed, performing a 'PUT' to "suppress\_account\_updater"="true" parameter would be required to pass for each card value.
- Disputes
  - Overview - The Disputes object represents transactions that were disputed by the credit card account holder. Dispute records are automatically submitted to Forte by the processor. Merchants can view these record and upload supporting documentation using the documents object via our REST API.
  - How it Works
    - Notifications - are available in Dex or sent via email.



TOGETHER, YOU WIN.

- If using Dex, log in credentials and user access rights required
- Responding – Responses can be leveraged within Dex or via REST API
  - Pulling/Requesting dispute records can be managed by Org or Merchant locations via, 'GET' request for all disputes for an Organization
  - Response calls require using the Documents object which enable merchants/partners to upload supporting documentation for disputes.
    - Documents must be less than 8MB and conform to message format.
    - Uploading documents require multipart/form-data content type, which includes a series of sections separated by boundary strings (alphanumeric up to 70 characters)
  - Get Disputes –
    - `{{baseURI}}/organizations/org_{{organizationID}}/disputes/`
  - Post Documents
    - `{{baseURI}}/organizations/org_{{organizationID}}/documents/`

### Credential On File Framework

- Overview – The Credential on File (CoF) Framework is a system defined by the card brands to aid issuers in identifying stored credential transactions allowing for differentiated (improved) treatment throughout the approval process. It provides greater visibility of transaction risk levels for issuers, higher authorization approval rates for merchants, and fewer customer complaints by way of improved cardholder experiences.
  - CoF Details
    - Stored Credentials are any payment instrument stored by a merchant or it's agent for the need to process future transactions on behalf of a cardholder
    - While there are several use cases, the utility/practice Boon Chapman will leverage primarily are the Standing Instruction requests, essentially a Merchant-Initiated Transaction (MIT). These can be
      - Installment Payments (I)
      - Recurring Payments (R)
      - Unscheduled Credential on File transactions -Variable (C)
    - Requirements include
      - Cardholder Consent
      - Data Field Flags/Indicators
        - POS Entry Mode Code (defines channel)
        - POS environment - Standing Instruction Indicator (C, R, I)
        - Original Transaction ID
    - Additional information can be sourced here
      - [Visa CredentialOnFileService\\_May10\\_4PM\\_FINAL.indd](#)
    - Additionally, Visa requires merchants that are not capturing payment immediately but choose to tokenize a payment credential for a future need is expected to process an Account Verification request (\$0 auth)



TOGETHER, YOU WIN.

## Recurring Transactions

- Overview – Boon Chapman has identified a need to support recurring transactions based on their operating model and business vertical needs. Essentially sending each individual transaction record as a ‘sale’ transaction. As defined above, there are card brand requirements when facilitating a MIT request. Alternatively, Boon Chapman does have the option to leverage CSG Forte’s Schedule endpoint to facilitate transactions.
- Schedules Object
  - Forte’s Schedules object captures scheduled transactions for organizations/locations. Those requests can be scheduled for a particular quantity or continuous until deleted or suspended. This method mitigates some of the requirements of the CoF framework. The Schedules object supports POST requests for new schedules, GET requests for existing schedules, PUT Requests to modify a schedule, and DELETE requests to terminate a schedule
  - Requirements
    - Scheduled transactions can only be created with customer/paymethod tokens
    - Actions supported are sale and credit
    - Frequencies include: One-time future, weekly, bi-weekly, monthly, bi-monthly, semi-monthly, quarterly, semi-annually, annually