

# Payment Apps Issues

WPWG FTF Meeting, March 2017

Ian Jacobs

# Summary

- Reuse of Web Technology
- Impact of Payment Apps on PR API
- PR API Data for Payment Apps
- Merchant Preferences

# Reuse of Web Technology

- Icons and labels for an app
- Permissions model (per origin)
- Service workers (asynchronous comms, ability to open windows)
- Lots more (e.g., authentication)

# Do we have what we need?

- Origin Model
  - Issue 98: Use cases for multiple services per origin: white label payment app, multiple user profiles (e.g., business v. personal).
  - Issue 94: Origin requests permission from user for payment app to handle payment requests.
  - Issue 105: What is a payment app (and relation to Web app)?
- Windows
  - Issue 97: Service workers defines an `openWindow()` but the behavior in a payments flow may not be appropriate (e.g., new tab on desktop)
- Icons and labels
  - Issue 69: Can use Web App Manifest for top-level label, icon, but we have finer grain requirements for Options and grouping

# Proposals

- Origin requests permission to handle payment requests.
- Origin describes Options and grouping at registration.
  - Can mix existing Web technologies with finer-grain Options via registration.
  - “Wallet” is a container
  - Question: Should user agents be required to display options (including cascading Options, e.g., on mobile)? Or can user agents in some cases only display top-level information and force payment apps to manage Options?
  - Question: How does merchant payment method order relate to payment app registration order? ([Issue 116](#))
- Propose new event.OpenWindow to improve window options for payment flows.
  - What happens to window when a problem arises?

# Impact of Payment Apps on PR API

- In September we asked whether integration of third-party payment apps would require changes to PR API.

# Do we have what we need?

- Support for push payment methods
  - requestId
- Support for matching merchant and payment app capabilities (issue 96)
  - Updated show() and canMakePayment()
- Native payment app integrations provides some assurances about PR API shape.
- Ten or so experimentations with Web-based payment apps; most using Tommy's polyfill.

# PR API Data for Payment Apps

- Should we reuse PR request/response structures as-is? ([issue 99](#)).
  - No, since only small subset of merchant data shared and other data added (e.g., origin)
- Should line items be shared with payment apps ([issue 91](#)), e.g., due to privacy?
  - No, if line items are used only for tax info, etc. [Resolved 16 March; Payment Handler updated](#)
- Should payment apps be able to request permission to access other **user data stored in browser**
  - Use cases include real-time credit computation (cf Klarna and developing economies).
  - Second use case of “easy enrollment” would likely require more data than available via PR API.
  - See [proposal from Frank Hoffmann](#) and [16 March discussion](#)

# Merchant Preferences

- In exchange for lower front end cost, merchants hand off some aspects of checkout to browser and payment apps.
- In exchange, can they make recommendations that could have an impact on the display of the user's registered payment apps? (Issue 74)
- Also, can they help bootstrap the payment app ecosystem through recommendations?

# Do we have what we need?

- Can merchants recommend payment apps?
  - Yes, in their pages as they do today.
  - We have decided against providing new (native) browser chrome to quickly access recommended payment apps due to security concerns.
- Can merchants express preferences and potentially affect display or ordering of registered payment apps?
  - We have not reached consensus.
  - Adding this would likely mean a change to PR API to allow the merchant to provide a list of URLs for payment apps.
  - Merchants can express payment method preferences already through PR API, and that information figures into ordering.

# Topics raised FTF

- How do apps expose functionality they support to optimize checkout flow (Alan)?