DAP Architectural Issues

Frederick Hirsch, co-chair DAP

3 November 2011, @fjhirsch

DAP Charter Quotes

- The mission of the
  [Device APIs Working Group is to create client-side APIs that enable the development of Web Applications that interact with device hardware, services and applications such as the camera, microphone, system sensors, native address books, calendars and native messaging applications.](#)

- This Working Group's deliverables must address issues of *accessibility*, *internationalization*, *mobility*, *security* and *privacy*.

Chartered Deliverables

- Calendar API, an API to access a calendar service (e.g. to add an entry, to edit an entry, to delete an entry)
- Tasks API, an API to access a personal task management / organizer service (e.g. to add, edit, delete a task)
- Contacts API, an API to access a contacts or address book service (e.g. to add an entry, to edit an entry, to delete an entry)
- Capture API, an API to manage a device's camera and microphone, e.g. to take a picture or record a sound
- Messaging API, an API to create and send a message. The API is agnostic to the underlying messaging service (e.g. e-mail, SMS, MMS).
- An API to discover the current network characteristics
- An API to react to a device power status
- An API to retrieve data generically from sensors
- An API that allows web applications to register themselves as handlers/providers based on data string identifiers and an API that enables other web applications to discover these handlers/providers and gain permission to interact with them.
- Gallery API, an API to manage the local media file storage
- An API to manage vibration
- An API to manage systems beeps
- An API to manage application menus
- An API to read the current audio volume level of a device
- An API for requesting and managing user permissions to use device features
- An API to discover and communicate with devices and services on the same device, on the local network, or directly connected, e.g. by USB and Bluetooth (*).
- Privacy Mechanism for Device APIs

APIs by Type

- Device focus
  - Battery, Vibration, Network Information
  - Discovery
- Information focus
  - Contacts, Calendar, Tasks
  - Gallery, Capture
  - Messaging
- Remote Data
  - Sensors

Chartered Deliverables

- Charter is carefully worded – does not say Javascript APIs, does not say device-only APIs.
- *However,* focus to date has been primarily
  - JavaScript APIs
  - Access to device specific resources
- Whether API is device specific or appropriate to web may depend on type of API
  - Information focus could be local or on web

Approach

- Discovery, permissions and privacy
  - Definition of mechanisms to find and bind to source of information

- API definition
  - Mechanism to manipulate information from bound source
  - Define using WebIDL
    - Currently defines JavaScript binding
    - Consider additional document for HTTP binding

Inputs related to architecture

- Powerbox/REST, ISSUE-94, July 2010
  - http://lists.w3.org/Archives/Public/public-device-apis/2010Aug/att-0032/minutes-2010-07-15.html#item07
- Web Introducer, March 2011
  - http://lists.w3.org/Archives/Public/public-device-apis/2011Mar/att-0097/minutes-2011-03-15#item07
- WebIDL REST Binding, March 2011
  - http://lists.w3.org/Archives/Public/public-device-apis/2010Mar/att-0154/minutes-2010-03-16.html#item04
- WebKit feedback on HTTP-based APIs, July 2011
  - http://lists.w3.org/Archives/Public/public-device-apis/2011Jul/att-0100/minutes-2011-07-21.html#item03

Next Steps

- Consider integration of discovery, security and privacy as web and device discovery mechanism

- Evaluate and progress HTTP binding to WebIDL

- Other approaches?

Thanks!

- DAP Home Page and Roadmap
  - http://www.w3.org/2009/dap/