



Delivery Context in Internet Protocols

Larry Masinter

Adobe Systems Incorporated



Relevant Background

- chair, HTTP working group ('95-'99)
- editor & participant in many other IETF standards groups:
 - Internet Fax
 - Instant messaging
 - Content negotiation
- Project lead, Xerox PARC System 33



Main points of this talk

- consider broader history & application for device context
- Content negotiation doesn't scale
- Vocabularies should be standards



History

- Fax: reliable messaging based on simple negotiation
- Printing, color management: history of adaptation
- System 33: early net service that did adaptation
- HTTP & various extensions
- Web: Cache control, management, sharing
 - most web traffic is to very small number of sites
 - popular web sites use content distribution networks
- Internet Print Protocol
 - Discover printer capabilities, formats supported, paper sizes
- Session Initiation (SIP)
- EMail negotiation
 - “send HTML mail or plain text”
- Instant Messaging
 - Send Images? Audio? Video?
not in standards proposals, but part of commercial IM



Side-note: Vocabulary

- sender: agent with content
 - receiver: agent that gets content
 - client: initiator, transient agent
 - server: accepts client requests, available agent
-
- In HTTP, client is receiver, server is sender, except in POST and PUT, which don't do content negotiation
 - Sender adapts content for receiver



History: Fax

- Sender connects to receiver
over telephone line
- receiver sends capabilities
*negotiates bandwidth, resolution,
compression methods*
- sender chooses format to match receiver
capabilities
typically before paper is scanned
- Standard mandates base mandatory-to-
implement format
insures negotiation will succeed



History: System 33

- Xerox PARC project, 1988-1994
- document storage & content adaptation
- Attempted to deal with devices of different capabilities, screen size, resolution, color capabilities
- Client sends preferences, server adapts
- Added conversion to HTML



Postscript Printer Descriptions

- capabilities (color space, memory)
- characteristics (resolution)
- initialization information
- font capabilities

Usage evolved over time



History: Color management

- **Calibration:** return device to known characteristics
- **Characterization:** create profile that describes condition of device
- **Matching:** convert given content to appropriate content for device
- **Gamut:** range of capabilities
- **Device Profile:** describes gamut and other characteristics of device
- **Generic device profile:** by manufacturer, for all devices of a given model
- **Custom device profile:** describes a particular device after characterization
- **Render intent:** choice of method used for matching, depending on the purpose



Content Negotiation in HTTP

- HTTP/1.0 RFC 1945 (May 1996)
- Accept (accept-charset, accept-language)
 - defaults?
 - pattern matching?
 - parameters?
- HTTP/1.1 RFC 1945 (Jan 1997)
 - q parameter
 - accept-encoding, accept-language



Proposed HTTP Extensions

- TCN: Transparent Content Negotiation, RFC 2295, March 1998
 - Accept-Features request
 - “Alternates” response
 - “Negotiate” request
 - “TCN” response
 - “Variant-Vary” response
 - Variant etags

Multiple modes for exchange of context information

Which features are not needed for functioning with content distribution?
- RVSA: Remote Variant Selection Algorithm, RFC 2296, 1998
 - no features registry (predated CONNEG)
 - Allow intermediaries to participate in content selection



Use today in HTTP

- “Accept” useless
- “Accept-Language” widely implemented, sent, but rarely used
- “Accept-Encoding” sometimes useful
- No interest in TCN or RVSA
- “User-Agent” most frequent use, but everyone is “Mozilla (Compatible)”
- Detection, when needed, done by JavaScript, Java “sniffers”



IPP Internet Print Protocol

RFC 2910, 2911

- Get-Printer-Attributes request
 - query for document-format, pdl-override, compression, job size, color capabilities, available printer drivers
- Print driver adapts user selectable job options, content, to match printer capabilities



SIP: Session Initiation Protocol RFC 2543 (and others)

- HTTP-like protocol for establishing multimedia communication (voice, video)
- OPTIONS, INVITE methods allow discovering capabilities
- Negotiate bandwidth, codecs
- proposed extensions for negotiating other preferences



EMail Extensions for Content Negotiation

- Address book
 - “send plain text or HTML?”
- Internet Fax
 - Use email-based image transmission
 - Message confirmation can indicate receiver capabilities
- VPIM: Voice Profile for Internet Mail
 - email-based voice messaging
 - Proposals to use IFax methods



EMail negotiation proposal

draft-ietf-fax-content-negotiation-05.txt

- Sender has limited possible cases
- Sender prepares and sends “standard” presentation
- Receiver may select “better” presentation and request it



RESCAP: resource capability protocol

- Create (DNS-indexed) resource capability services
- Include device capability, public keys, protocol capabilities, etc.
- Not moving very fast




CONNEG: vocabulary for media features

- Started ~1996 out of HTTP working group
- Working group given narrow charter: create vocabulary for content negotiation started with HTTP use case
- Create “Media Feature Registry”: RFC 2506
- Define common media features: RFCs 2534, 2912, 2987
- Create syntax for media feature expressions RFCs 2533, 2738
- Shorthand for expressions: RFC 2938



CONNEG basics

- Registered features:
 - Display, print & fax:
pix-x, pix-y, dpi, ua-media, paper-size
 - MIME type, charset, language
 - more elaborate capabilities for fax
 - proposed feature for xmlns
- Compact syntax for boolean expressions:
(| (& (pi x-x=750) (pi x-y=500) (color=mapped))
(& (dpi=300) (ua-media=stationery)
(papersize=iso-A4)))
- Hash canonical syntax for references
(& (dpi=100) (h. SBB5REA0MHC09CP2GM4V07PQP0))
where
(h. SBB5REA0MHC09CP2GM4V07PQP0) : -
(& (pix-x<=200) (pix-y<=150))
end



Warning: different meanings for same vocabulary term

- Capabilities:
“I can display up to 300 dpi images”
- Characteristics
“I have a 300 dpi screen”
- Preferences
“I prefer images an integer fraction of 300 dpi”
- Content’s characteristics
“This image was scanned at 300 dpi”
- Content preference for device capabilities
“This image best viewed on a 300 dpi display”



Standardize vocabularies

- listing, registering, using URIs for vocabulary terms not enough
- Vocabularies should be standards, with demonstrated interoperability
- Must include clear definition of interpretation, allowable content, implications for adaptation



Avoid content-negotiation failure

- origin-server adaptation doesn't scale
- origin-unaware adaptation works poorly
- Many other protocols are too complex for deployment
- Receiver-makers: *"best viewed by" me!*
content authors hate it
- want author-once, view many
 - Reuse & retransmission (forward, print)
fail with adapted content
 - signatures & version management difficult



Device-Independent Content

- Device-independent content may be “multi-modal”, self-adapt to context
- **Embedded vocabularies** in scripting language, media queries, need to be standards