Overview

- Mobility Landscape
- Challenges of Mobility
- Example
- Development Approaches
The Mobility Landscape

- W-ISP
  - Device: Browser
    - Digital Phones
    - PDA
    - 2-way Pager
    - Windows CE
    - Symbian EPOC
    - Stinger
    - Palm OS
  - UP
  - Nokia
  - AUS Browser
  - Pocket IE
- W-ASP
  - Gateway
  - Internet
  - cHTML
  - xHTML
  - miHTML
  - sHTML
  - HTML
  - XML
  - HDML
  - WML
  - SMS
  - SMTP
  - Over
  - HTTP
  - HTTP/S

Enterprise
  - User Experience Server
  - EIP
  - EAI
  - App Server
  - CRM
  - Database
  - SQL
  - Oracle
  - DB2
  - Java
  - COM
  - JDBC
  - ODBC
  - LDAP
An Extension of the Existing Internet Infrastructure

- The same protocols (e.g. HTTP)
  - WTP/S (the transport protocol of WAP) is typically translated to HTTP/S by the gateway
  - Most protocols are coming back to HTTP, the wireless specific ones are network level below TCP/IP

- The same markup languages (e.g. HTML)
  - Most devices are converging towards XHTML

- Think of Mobile Devices as being equivalent channels to the wireless internet
But there are big differences...

- Device form factors, not markup languages, drive the user interface
  - Different devices can drive totally different data and usability

- You need to test across many browser/markup combinations
  - There are no less than 175 combinations

- The differences are not trivial and often require complex design and development
  - e.g. On a phone call out to phone number for a contact page
The Market Evolution

- Exploding diversity of devices and standards
- GUI must adapt to device, language, etc.
- Device specific vs. device agnostic mobile experience
- Personalized and integrated experience, not monolithic apps
- Seamless continuity between work and home
Challenge: Device Proliferation/Diversity

Devices, Form Factors and Browser Proliferation

- Device proliferation
- Users have diverse devices
  - Users will use multiple devices to access application (Gartner group predicts 3-4 devices)
- Browser Proliferation
  - Support multiple browsers (not 2 as in web world)
  - Multiple language support (wml, hdml, html, chtml, ..) with multiple version support
- Hardware capabilities
  - CPU, Memory, Battery life
- Online, Offline and Voice
Differing Device Form Factors

- Simple Internet-enabled Phone
- Smart Phone
- Pagers
- Personal Digital Assistant (PDA) Numeric keypad used to enter numbers, text and special characters. Special keys available to provide additional input.
Differing Device Form Factors

- Simple Internet-enabled Phone
- Smart Phone
- Pagers
- Personal Digital Assistant (PDA)

Numeric keypad or miniature keypad used to enter numbers, text and special characters.
Differing Device Form Factors

- Simple Internet-enabled Phone
- Smart Phone
- Pagers
- Personal Digital Assistant (PDA)
Differing Device Form Factors

- Simple Internet-enabled Phone
- Smart Phone
- Pagers
- Personal Digital Assistant (PDA)

Stylus for clicking, writing (character recognition) and tapping on a virtual keyboard. Some accept voice input.
**Difference in Device/Browser Behaviour**

**Link Labels**
- Most UP phones cut labels off at 6-7 characters for display
- Qualcomm 2760 will not display numerals in link label
- On some Samsung Phones large right link labels are displayed over left link labels

**Titles**
- Title text is cut off on must WAP devices, titles are not wrapped (exception UP.Browser V4)
- Ericsson will display URL if no title is specified
- UP.Browser and Go.Web browser can’t display titles containing variables

**Variables**
- Nokia 7110 and AUSWAP browser fail if variable names start with numbers
- Most WML devices do not support required variables and variable masking
- Neopoint (UP.Browser) does not perform variable masking correctly
Developing the GUI

1. Perl/ASP/JSP/etc.: Develop separate GUIs for each device
   - Pro: Familiar technology
   - Con: Too many pages to develop

2. XSL stylesheets: Develop separate XSL stylesheets
   - Pro: Exploiting XML/XSL, have fine grained control of GUI
   - Con: Still have many stylesheets to develop

3. Transcoders: Products automatically translate pages
   - Pro: Develop once with
   - Con: The GUIs are unusable, no fine grained control

4. XML-based mobility products
   - Pro: Target many devices with minimal programming
   - Con: Many rely on a proprietary development language
Using XML/XSL

To use an XSL based solution you need:

- A processor to match the device and choose the right stylesheet
- A processor to get the appropriate XML data and call the XSL interpreter
- Some post processing to take care of encoding issues with different devices
The Configuration Management Headache

- Writing all those stylesheets
- Testing all those devices
- Ooohh all those stylesheets
- Managing the links
- Dealing with pagination
- The exceptions
- Code page translations
Tools can Help

- **XSL/XSLT Editors**
  - Support tools for authoring

- **EAI/EIP Vendors**
  - Legacy system access and integration

- **Mobile Platforms**
  - Frameworks for the unique challenges of Mobility
Conclusion

- Mobility is really an extension of the existing infrastructure rather than a replacement.
- The big difference is in the form factors and GUI choices, not the protocols/markups.
- Standards like XML and XSL can help solve the problem of targeting many devices and still allowing customizations and tweaks.
- There are many headaches in managing the XML/XSL for which freeware and packaged products exist.
Thank You

David R. Seaman
VP Product Development
Viafone Inc.
dave@mobileq.com
http://www.mobileq.com