Context dependent services in an m-learning environment: the printing case

Anna Trifonova & Marco Ronchetti
Dipartim. Informatica e Telecomunicazioni
Università di Trento
Italy

M-learning

- There is a common agreement that m-learning is e-learning through mobile computational devices.
- In general by mobile device we mean PDAs and digital cell phone, but more generally we might think of any device that is small, autonomous and unobtrusive enough to accompany us in every moment in our every-day life, and that can be used for some form of learning.

M-learning: when is it useful?

- **5 minute value:** ability to use small fragments of time for learning
- **Simplicity:** limited display and input capabilities Þ not practical to transpose a power-point presentation on a PDA;
- **Context dependent information:**
  - *location context* ; the system knows the location where the learner resides and adjusts to it;
  - *temporal context;* the system is aware of time dependent data;
  - *behavioral context* ; the system monitors the activities performed by the learner and responds to them adjusting its behavior;
  - *interest specific context:* the system modifies its behavior according to the user's preferences.

M-learning architecture

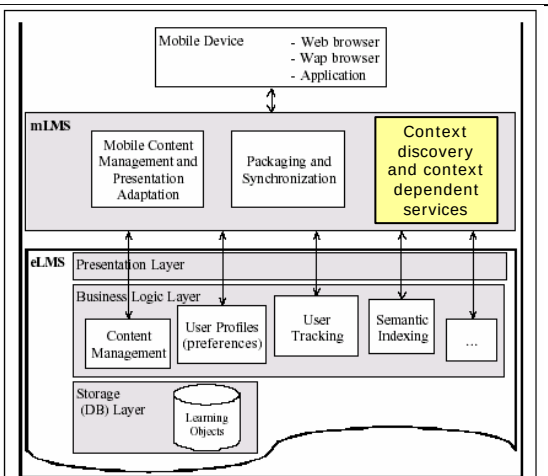
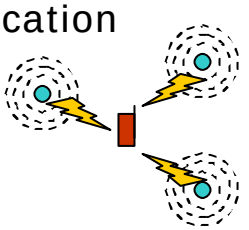


Figure 1: General and Generic m-learning Architecture

Positioning systems

- Ad-hoc
  - Outdoors
    - Global Positioning System (GPS)
  - Indoors
    - infrared, ultrasonic, radio
- General (using communication infrastructure)
  - GSM cell
  - Wi-Fi signal strength



Positioning systems

- "raw"

  e.g. latitude/longitude/height
- semantically meaningful

  e.g. "Science Faculty Building, Office D11, 2nd floor"
- An architectural layer could be delegated for the conversion

  See e.g. J.Roth's work

Context dependent services in m-learning: a sample problem.

Print on the nearest available and suitable printer

- *It's a prototypical problem*
- *Has many features*
- *Has many added problems*

Printing from a mobile device.

- Can depend on
  - Location
  - Time
  - User identity (permissions)
  - User preferences
  - User activity
  - Status of resource
- happens *from any application*
- uses services that are provided by the (local) *operating system* and must be made remote
- *Resource accounting* may be needed
- Needs to scale

Actors

- The mobile user (PDA)
  - Requests a *service*
- The location server
  - Gets row signal data, gives location
- The discovery/proxy server
  - Knows where *services* are available, can redirect or act as a proxy (spooler)
- The authorization server
  - Authorizes requests

Note: We do not talk of printing!

The process: 1 - Discovering location

- PDA looks for location server (LS)
  - DHCP-like interrogation
- PDA sends row signal data to LS
- LS sends back location information
  - Either raw or cooked

The process
2 – printing (version 1:find info & print)

- PDA looks for printer server (PS)
  - DHCP-like interrogation
- PDA asks PS for a suitable printer
  - By passing all needed context data
- PDA prints on the printer

- PDA needs a printer driver!
  - Download + install?
    - Invasive
    - OS-dependent
    - Rebooting might be needed!
    - Little control on authorization

The process
2 – printing (version 2.1 :spool file)

- PDA looks for printer server (PS)
  - DHCP-like interrogation
- PDA sends file to PS
  - Along with context data
- PS prints on the printer
- PDA gets info from PS about used printer

- PS needs to have all apps!
    - cumbersum
    - OS-dependent

The process
2 – printing (version 2.2 :spool dif)

- PDA looks for printer server (PS)
  - DHCP-like interrogation
- PDA prints on a virtual device (e.g. pdf, postscript)
- PDA sends device-independent-file to PS
  - Along with context data
- PS prints on the printer
- PDA gets info from PS about used printer

- It's a two-step process
- Can be unified by writing a suitable pseudo-device driver

Our first implementation

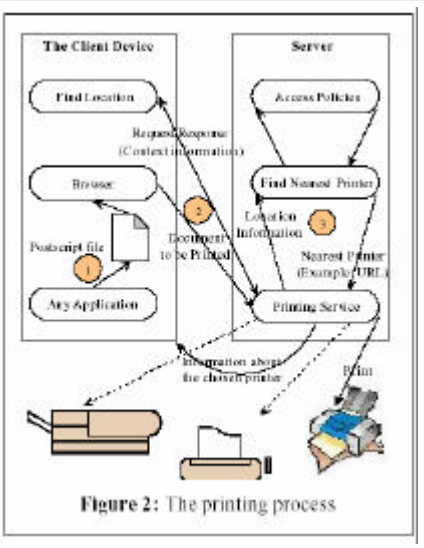


Figure 2: The printing process

A two-step, browser based process

The process:
3.1 – Authorization-accounting

Before printing:

- PDA sends to PS a signed request
  encrypted with its private key, with an indication of its authorization server (AS)
- If PS trusts AS:
  PS sends request to AS and (if credentials are ok and AS trusts PS) gets back authorization
- else:
  PS sends request to a trusted broker, that forwards request to AS
  - Broker acts like a credit card company

The process:
3.2 – Authorization-accounting

After printing:

- If PDA has to complain, sends notification to both PS and AS
  - PS should not charge service, or can signal a problem to AS
  - AS should not accept charge, or should take trace of the problem, to distrust either PDA or PS in future.

Università degli Studi di Trento

Conclusion

- We have examined the printing problem from a mobile device as a prototype for a class of problems arising in m-learning.
- We have proposed a solution that scales, and that can be mapped on other context-dependent problems.