Generating mobile device user interfaces for diagram-based modelling tools

Dejin Zhao
School of Information Sciences & Technology
Penn State University
dzhao@ist.psu.edu

John Grundy & John Hosking
Dept of Computer Science
University of Auckland
New Zealand
{john-g,john}@cs.auckland.ac.nz
Talk Outline

- Background
- Why mobile deployment?
- Constraints and Requirements
- Solution Architecture
- User Interface Adaptations
- Implementation
- Evaluation
- Summary
Our research focus: Meta tools for specifying and generating multiple view, multiple representation diagrammatic tools

- JViews/JComposer (Java heavy weight UI)
- Pounamu (Java, light weight UI)
- Marama (Eclipse platform specific)

Typical use: specification of domain specific visual language environments
Background
Background

- Want to be able to deploy generated tools on a variety of platforms
  - Thick client
  - Thin client (browser based)
  - Mobile devices (thin client, low resolution, low bandwidth)
- With no additional programming needed by the tool designer
### Why mobile deployment?

- Increasing need to review and amend diagrammatic information while mobile
  - Particularly useful for:
    - *Project management applications eg Gantt charts*
    - *Design/installation/maintenance diagrams for on site use*

- Increasing convergence of mobile functionality onto one handheld platform
  - Corresponding unwillingness to carry multiple devices around

- Increasing size and resolution of handheld device UIs
**Constraints and Requirements**

- **Constraints imposed by mobile device deployment:**
  - Relatively low processor speed, small memory and storage
  - The wide range of devices with a variety of different operating systems
  - Diagram rendering limitations of most current mobile devices

- **Requirements resulting from these constraints:**
  - Need for techniques to display large diagrams on small screens and yet still keep the diagram meaningful
  - Need for techniques to navigate large diagrams and between multiple views (diagrams) of a model
  - Need to support user preferences so that different users can specify different diagram content rendering, zooming and navigation configurations via their mobile device
Mobile Deployment Architecture

User devices:
- MUPE Client Browser
- JAVA MIDP 2.0

From Nokia

Pounamu MUPE Server:
- Property sheet generator
- Model UI generators: diagram and text views
- tool configuring
- Model data info from Pounamu
- Users’ tool configurations (XML)

Request handlers

Pounamu Host:
- RMI API
- Model views
- Models of tools
- Meta-tool specs (such as UML tool, project mmt tool)

Pounamu XML

MUPE XML
Example usage

Overview views use proportional diagram shrinking & omit details

Seperate button accessible menu
Goal is to eliminate additional programming
  - Aim to directly generate diagrams from same Pounamu XML spec as for thick client

But direct representation of complex diagrams on mobile devices is problematic
  - Screen size and resolution
  - Navigation and selection difficulties

Thus need some adaptations but need these to be generic so no programming required, just minimal end user configuration

Specific adaptations:
  - Multiple configurable levels of detail for diagram elements
  - Navigation/zooming support
  - Editing support
  - End user configuration support
Levels of detail

- Allows users to define multiple representations for a diagram at different levels of detail.
- Each diagram element can be separately selected and zoomed between its multiple levels of representation.
- Automatic zooming of elements is supported as users navigate a large view.
Navigation/zooming

- Mobile phone arrow keys can be used to navigate between elements
  - Selected element highlighted and status info shown at bottom
- Hot key used to zoom selected element between levels of detail
- Auto zoom magnifies selected element and surrounding elements
  - Rudimentary distortion oriented display
- Pan navigation provides floating panel on overview view which is used to select where to pan to
Limited interaction capability of mobile devices require additional adaptation for editing diagrams

Direct manipulation of elements replaced by 2 step selection & modal modification
- Elements moved via direction keys
- Elements added by narrowing pan selection region to show place to add

Element properties edited via separate form-based property sheet
- This is the major editing need
- Hot key selected
End user configuration support

- Principal support is for specifying level of detail representations
  - Shape, size, colour of icons by diagram and icon type
  - Properties shown
## Evaluation

- Limited user evaluation undertaken
- Viewing capability compares well to thick client
- Zoom features found to be essential for understandability
  - Users suggested these features might be useful for thick client
- Editing awkward but fine for most common operation of property editing
- More than 15 iconic elements proved problematic
- Automatic layout of diagrams would be useful
  - Contrary to thick client
  - Due to limited placement controls on mobile devices used
Conclusions

- Have demonstrated automatic generation of diagram-based editing environments for deployment on mobile devices
  - Generic – works for any Pounamu generated tool
  - Integrates a set of user adaptations that together mitigate screen resolution and interaction limitations
    - None are particularly novel, but their integration is

Future work

- Generalising from the mobile and thin client interfaces we have developed to provide a more general adaptation framework
  - Apply to other interfaces, eg 3D
- Port our work to our new Eclipse-based Marama meta tool
  - Thin client and mobile interfaces for Eclipse tools