Mobile App Discovery through Conceptual Models

Jasmin Brakmic, BSc

Supervisor: Prof. Dr. Dimitris Karagiannis
Research Group Knowledge Engineering

University of Vienna, 24.01.2013
Agenda

1. Motivation
2. Objectives
3. Related Work
4. Design and Implementation Framework
5. Master thesis roadmap
Motivation

• Growing trend in the development of mobile applications.
  – Google Play Store & iOS App Store -> over 600,000 mobile apps.
  – Nokia Ovi Store & Microsoft Store -> over 120,000 mobile apps.
• Mobile apps currently mainly developed from scratch -> time consuming and resource intensive development process.
• Implementation specific information cannot be accessed by developers and often does not fit the requirements for functionality because of inadequate search mechanisms.
Objectives

• Design modelling support for:
  – description of mobile app capabilities
  – app code selection and reuse

• Implement:
  – an app discovery web service as proof-of-concept
Related Work - Technological-oriented

• Regarding app implementation:
  – Guidelines for various platforms (Android, iOS, Windows Phone, etc.)

• Regarding mobile app functionality description:
  – UIML (User Interface Markup Language)
  – UsiXML (USer Interface eXtensible Markup Language)
  – XIML (eXtensible Interface Markup Language)

• Regarding web service description and discovery:
  – SOA technologies: UDDI, WSDL
Related Work - Theoretical-oriented

- Regarding mobile app discovery focused on deployed apps:

- Regarding mobile app design:

- Related to modelling and modelling methods:

- Regarding semantic description of web services:
FRAMEWORK
Framework inspired by the Publish-Bind Model for Web Services

- **Service consumer**
- **Service provider**
- **UDDI (Service Directory)**

**App developer** (as consumer) uses service, finds service, registers service, and requests to model-based search for app code.

**App directory** registers app code and allows model-based search for app code.

**App developer** (as provider) reuses/adapts app and registers app code.
Proposed procedure and architecture

<table>
<thead>
<tr>
<th>Domain expert</th>
<th>Provider</th>
<th>is described by</th>
<th>Consumer</th>
</tr>
</thead>
<tbody>
<tr>
<td>User Interaction specific models</td>
<td>Identifies domain</td>
<td>Process model</td>
<td>Identifies domain</td>
</tr>
<tr>
<td>Web service interface</td>
<td>Creates model and app</td>
<td>App models</td>
<td>Creates models</td>
</tr>
<tr>
<td>Web logic</td>
<td>Registers model and code</td>
<td>Queries</td>
<td>Uploads models as queries</td>
</tr>
<tr>
<td>Repository</td>
<td>Stores model and code</td>
<td>Search algorithm</td>
<td>Reusable/Adaptable code</td>
</tr>
</tbody>
</table>

- Process model
- App models
- Queries
- Search algorithm
- Concept-based structure
- Reusable/Adaptable code

- Assumed steps
- Supported steps
- To be implemented/defined
- Reused from ComVantage

Jasmin Brakmic, BSc.
Areas supported and described by App Models

App Capability

Preconditions

Effects

App Choreography (user interaction)

App Orchestration (interapp interactions)

App Information Resources

Will be used from the ComVantage Project.

Jasmin Brakmic, BSc.
Queries

- As an input for queries the Process model and App models will be used.
- Query examples based on:
  - **Capabilities**: Find me an app which provides authentication and book buying capabilities.
  - **Information resources**: Find me an app which produces as output a bill.
  - **App structure**: Provide me with an app which takes as input an username and password and sends the information to a server.
Search Algorithm

- The search algorithm will consider and extract the information stored in the conceptual models and try to find implementation specific information from the repository.
- Because of the diversity of the conceptual models for each model different comparison approaches will be used.
- Comparison will be based on:
  - Capabilities and Information,
  - Mobile App structure,
  - Mobile App Orchestrations (in progress).
Proposed Comparison Approaches for Conceptual Models

• Comparison based on Capabilities and Information resources -> Jaccard similarity coefficient.

\[ J(A, B) = \frac{|A \cap B|}{|A \cup B|} , J(A, B) \in [0, 1] \]

• Comparison based on two mobile app structure-> Pearson Correlation Coefficient.

\[ r_{ab} = \frac{n \sum a_i b_i - \sum a_i \sum b_i}{\sqrt{n \sum a_i^2 - (\sum a_i)^2} \sqrt{n \sum b_i^2 - (\sum b_i)^2}} , r_{ab} \in [-1, 1] \]
Mobile App Repository

- Is aimed to hold implementation and conceptual specific information about mobile apps.
- The repository should not be restricted to a certain implementation type (e.g. Android).
- Challenges for structure and interface:
  - Semantic enrichment of mobile app search?
  - Storage structure?
  - Mobile app queries?

Search algorithm expected results

- The search algorithm should not only perform querying on one mobile application, instead it should consider the best fitting combination of mobile application components of different mobile applications.
- The user should be provided with a possibility to define the number of best results, which should be returned.
- Example:
Roadmap

1. Describe mobile apps with ComVantage MM
2. Identify concepts for conceptual modelling of mobile app capabilities
3. Define structure for the app repository
4. Define query interface (input and output structure)
5. Define a search algorithm on the app repository.
Thank You For Your Attention!

Jasmin Brakmic, BSc
*University of Vienna*
Faculty of Computer Science
Research Group Knowledge Engineering
Währinger Straße 29
A-1090 Wien
jasmin.brakmic@univie.ac.at