A Blackboard-Like Architecture for the Development of Evolving High Fidelity Mobile Application Prototypes

– work in progress –

PID-MAD
@MobileHCI
27 August 2013

B. Igler · T. Braumann · S. Böhm
RheinMain University of Applied Sciences Wiesbaden

Motivation:
Identification of Key Success Factors

Success Factors of **M**obile **A**pplication Design for Public **T**ransportation

success factor = **feature** that **matters**

- ➡ feature-based evaluation & development
- ➡ lab and field tests
- ➡ evolving prototypes (low → high fidelity)
- ➡ high level of variability

Background and Related Research

- Process models with a focus on user satisfaction
  - agile process models
  - user-centered design

[Martin 2012]
[Gulliksen 2003]
[Humayoun 2011]

- Feature-based development of mobile apps
  - Software Product Line Engineering
  - main focus on different platforms

[Clements 2002]
[Quinton 2011]

- Flexible component models
  - service-oriented (heavy-weight)
  - agent-based (heavy-weight)

[OSGi]
[Padovitz 2008]
[Shaw 1996]

➡ Research Opportunity: simple component-structure for evolving mobile HiFi Prototypes

Proposed Approach

Major Requirements

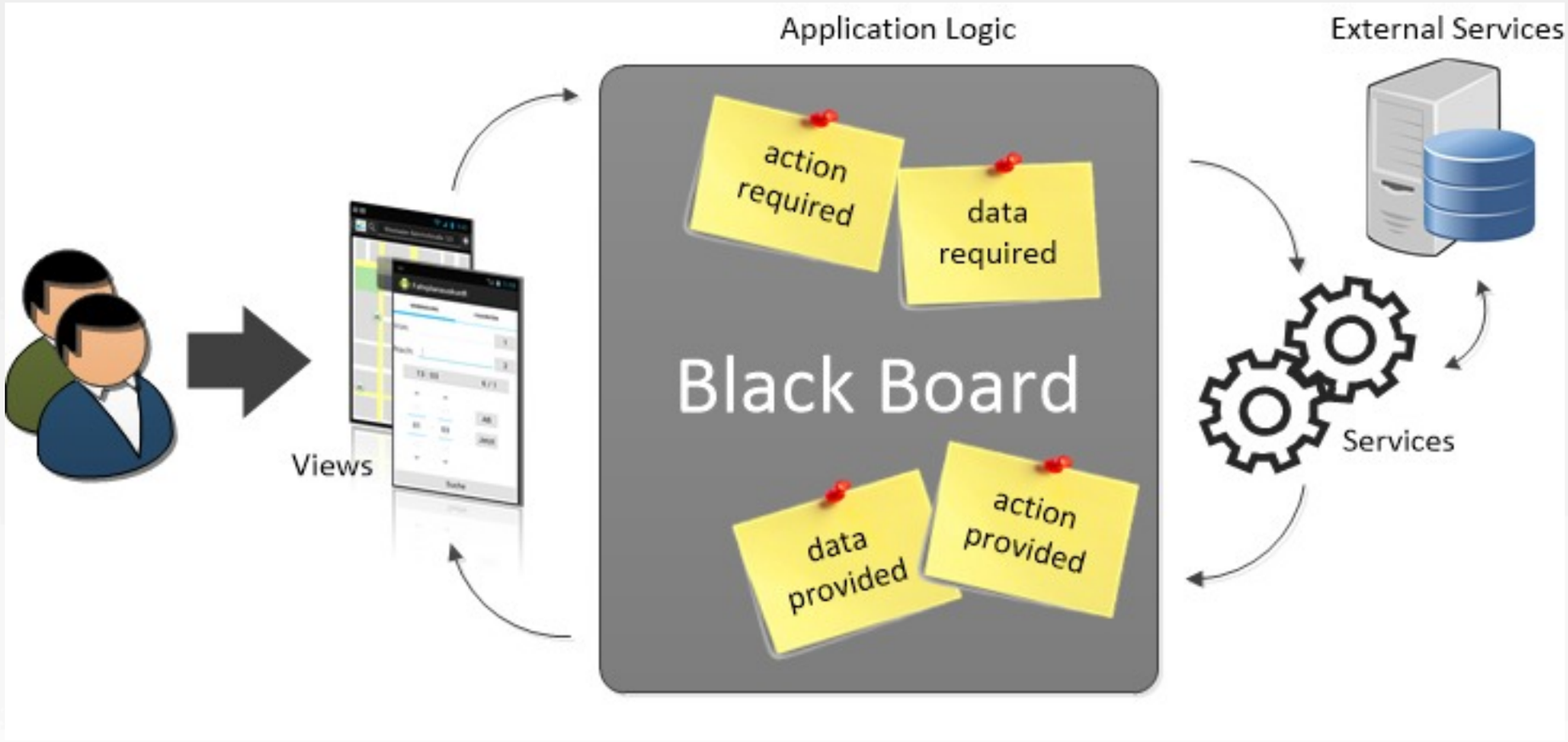
- high fidelity (ui and application logic)
- evolving prototype family (many changes/variants)
- low development effort
- simpler than e.g. agent-based approach

Proposed Prototype Structure

- family of evolving prototypes = growing set of components
- components
  - independent
  - interchangeable
- components cooperate via blackboard

(Adapted) Blackboard Concept

Application: set of independent, cooperating „experts“
Expert: view or service
Cooperation: posts on blackboard



Example (Cooperating Views)

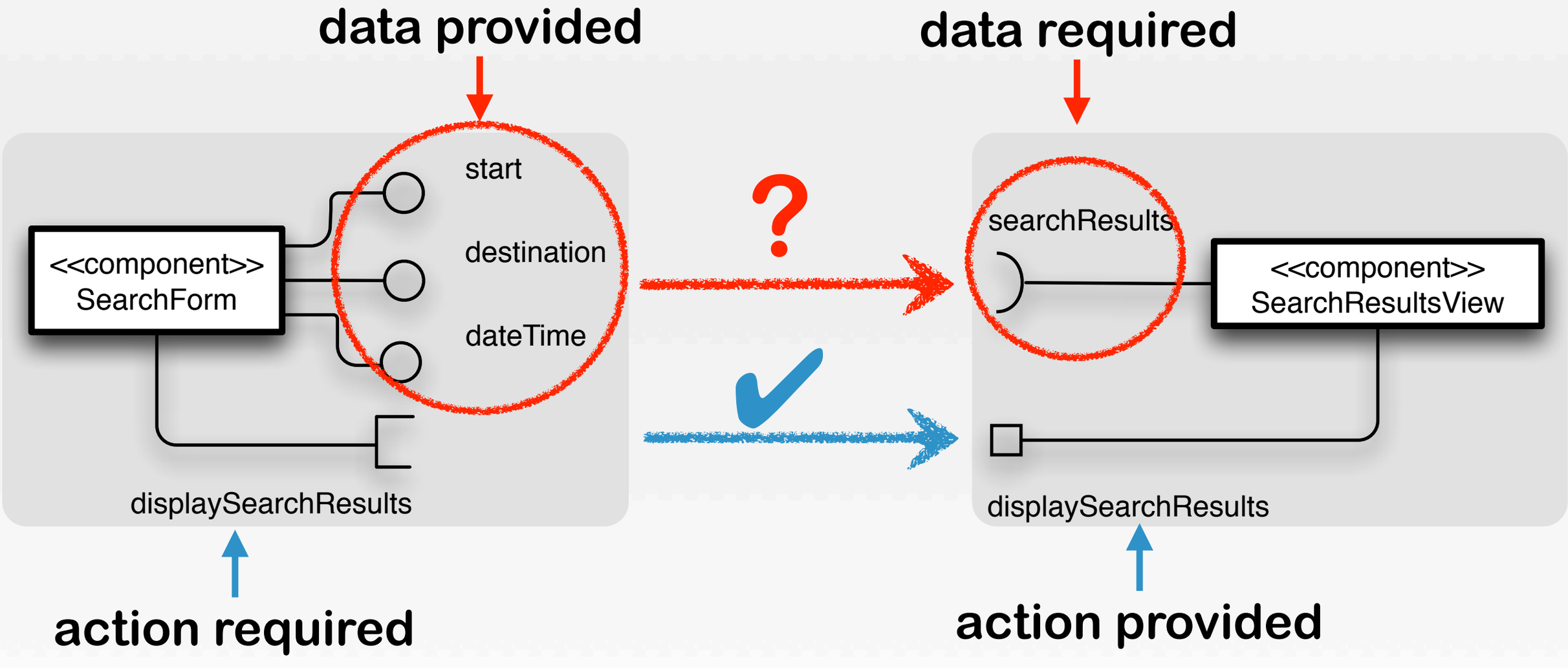
Search Form

Search Results

Example (Component Model)

Search Form

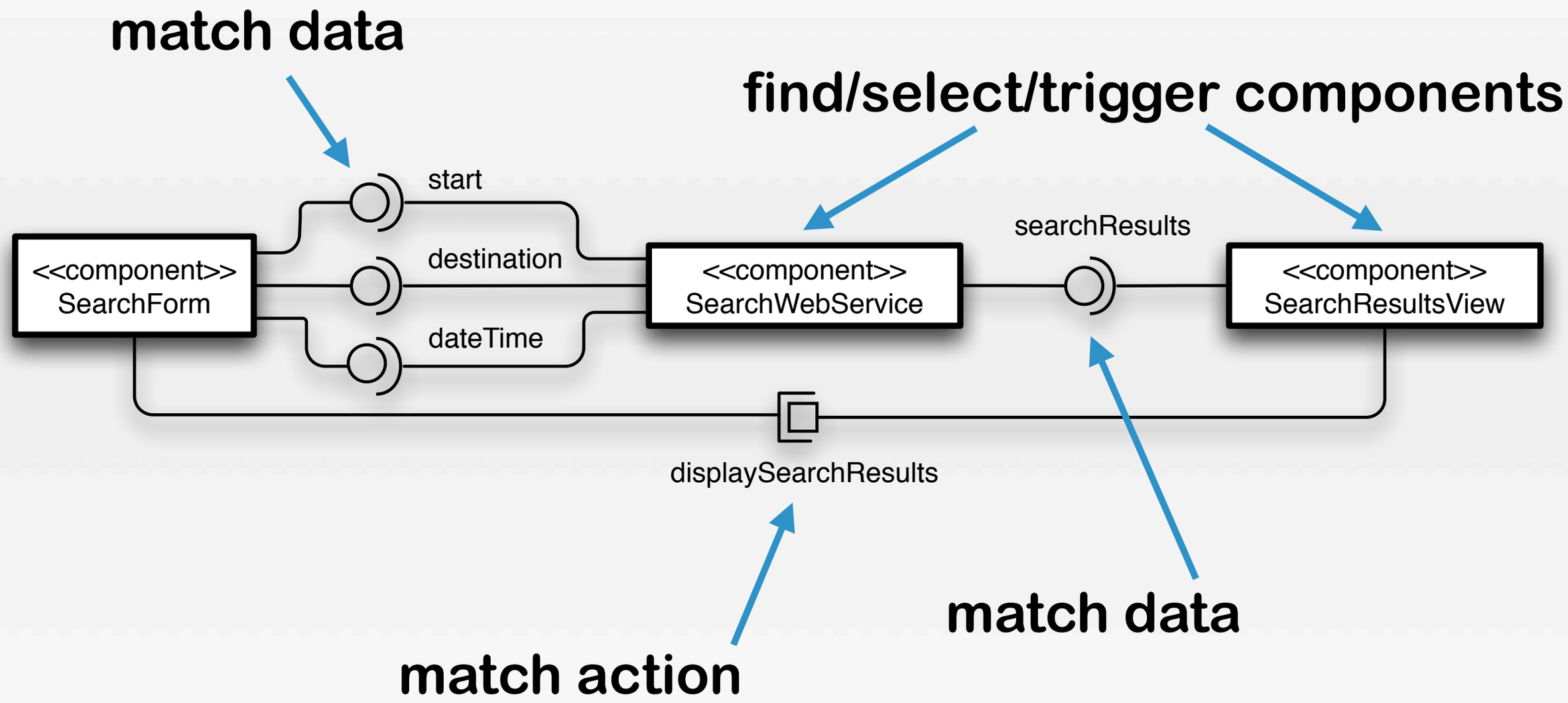
Search Results



action required

action provided

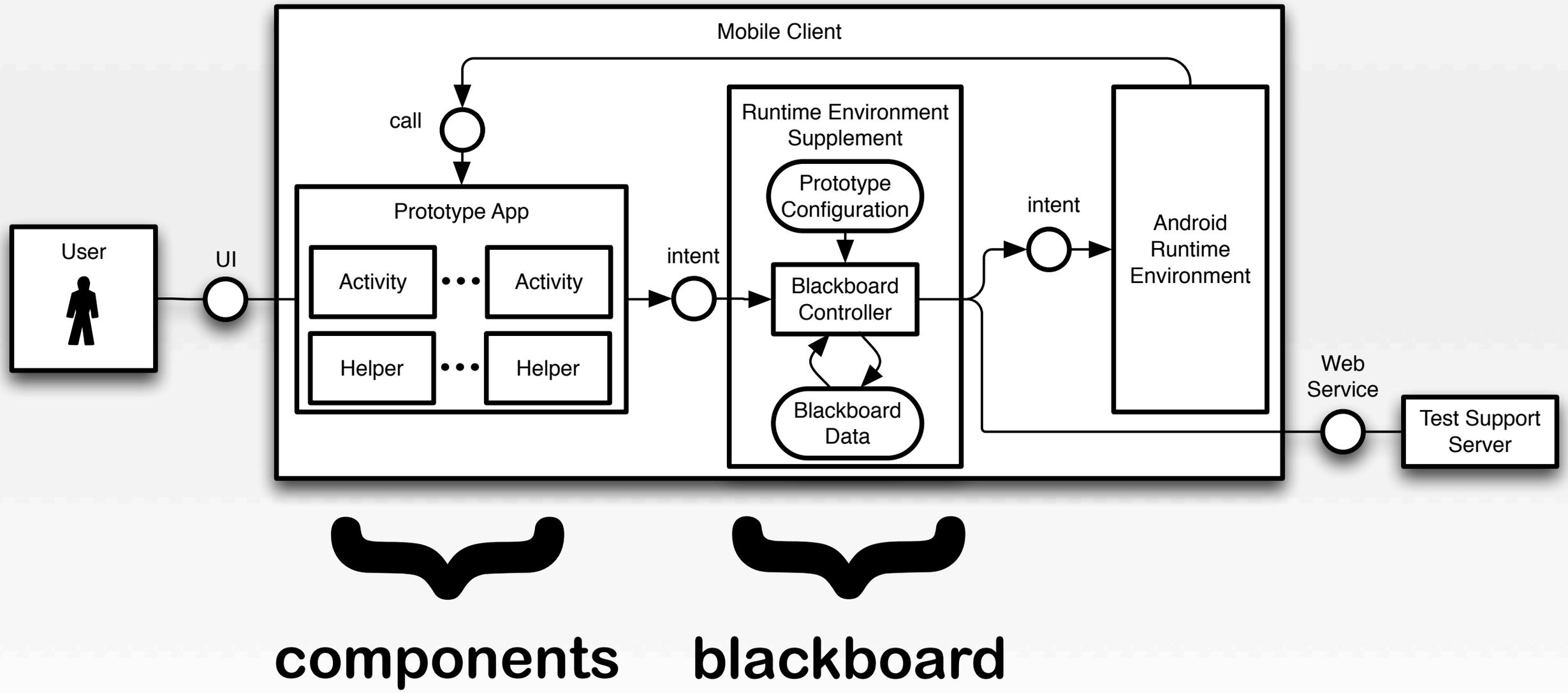
Example (Responsibility of the Blackboard)



In general: Routing/resolution can be recursive.

Proof of Concept: Blackboard-Framework for Android OS

Software Architecture (Outline):



Experience

Test-run with several iterations:

- **lo fi → hi fi prototypes**
- **iterative-incremental**
- **proprietary software dev. environment**
- **usability evaluation**
- **final iterations: end-user focus groups**

Conclusion

- The prototypes are fit for lab and field tests.
- The prototype family covers the whole feature variability range, including hifi features.
- The implementation effort for feature changes is generally low.
- Complex routing and insidious tight coupling complicate prototype development.

Future Directions

- Framework enhancements
  - for developers
  - for test managers
- Incorporation of usability evaluation features into framework

Discussion, Questions, Answers