Design of a Component-Based Augmented Reality Framework

Martin Bauer, Bernd Bruegge, Gudrun Klinker, Asa MacWilliams, Thomas Reicher, Stefan Riss, Christian Sandor, Martin Wagner

Technische Universität München
Institut für Informatik

Summary

- We think the time has come for augmented reality frameworks
- This will allow software components to be reused between different AR applications
- We have built and tested a first version of such a framework, called DWARF

Outline

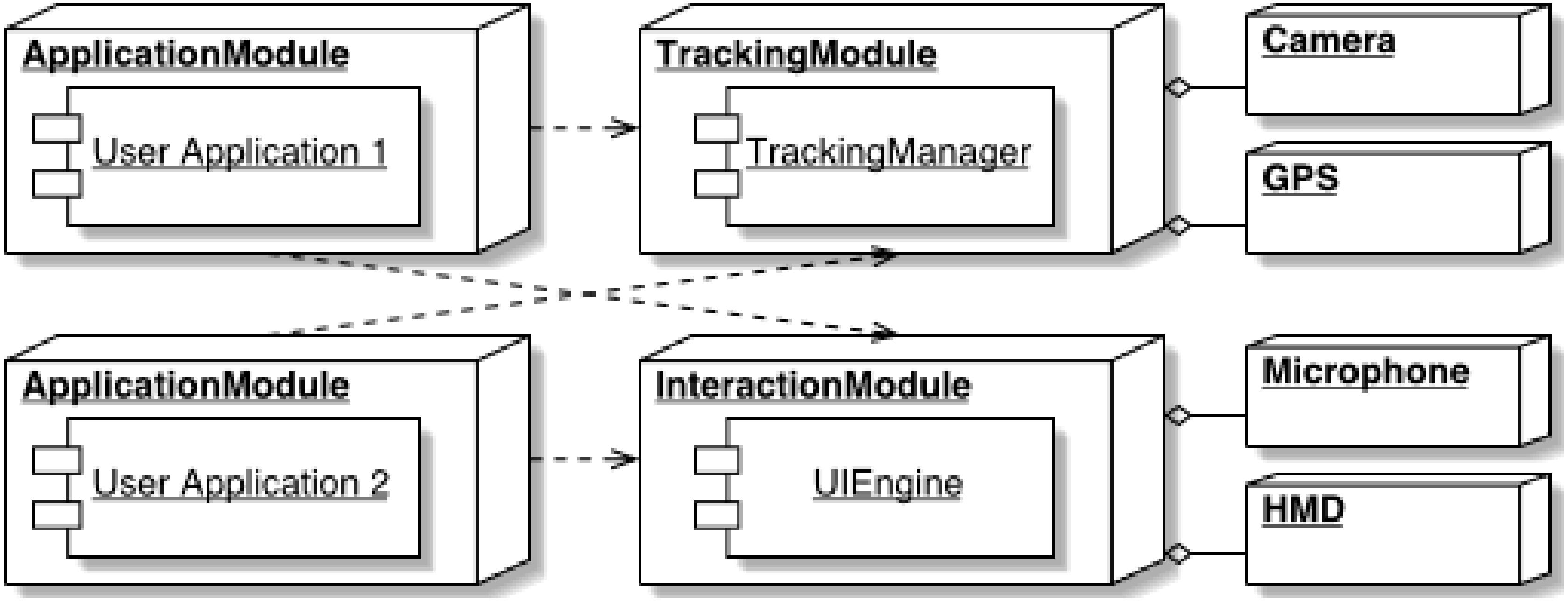
- Advantages of an AR framework
- Necessary elements of an AR framework
- DWARF – architecture and components
- Demonstration system

Advantages– Project manager

- Components can be reused in different applications
- Components are “black boxes” for other components
- Distribute development in time and space
- Rapid prototyping of AR applications becomes feasible

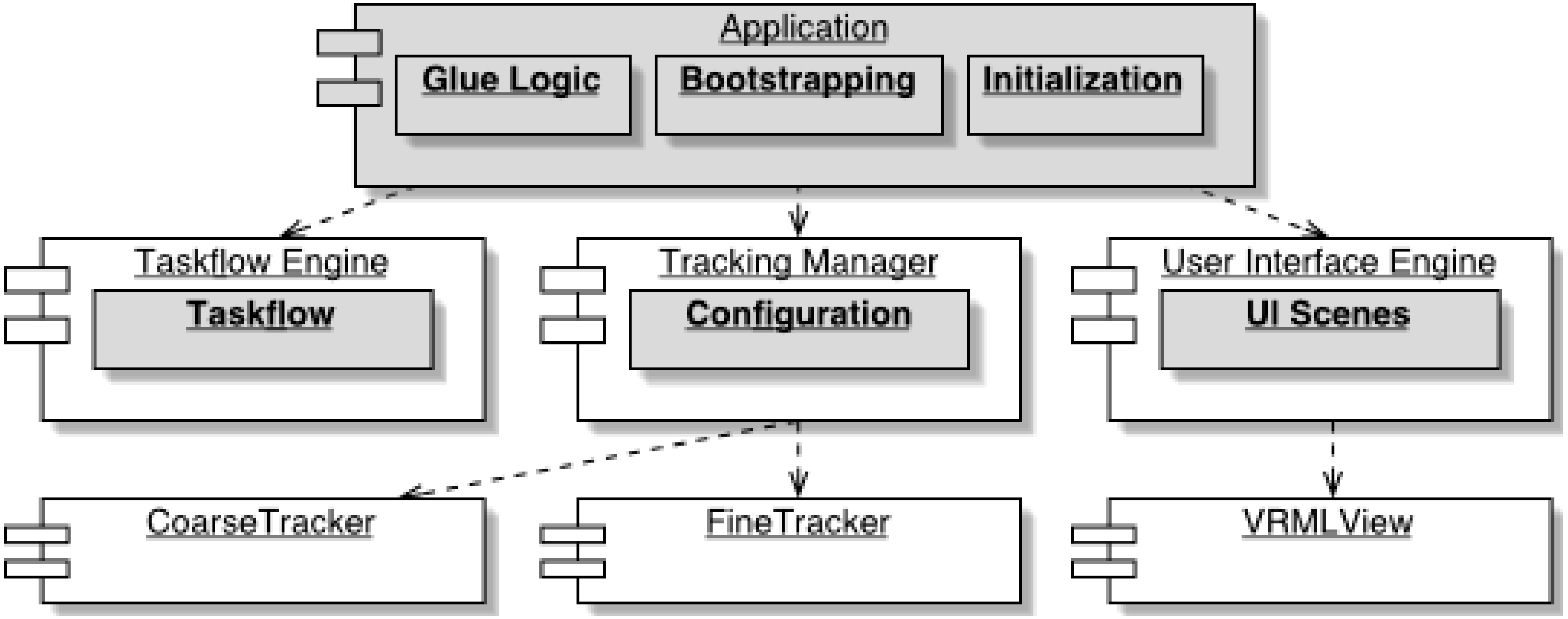
Advantages– User

- Software is integrated with hardware in wearable or stationary *modules*
- System is reconfigured by re-plugging modules



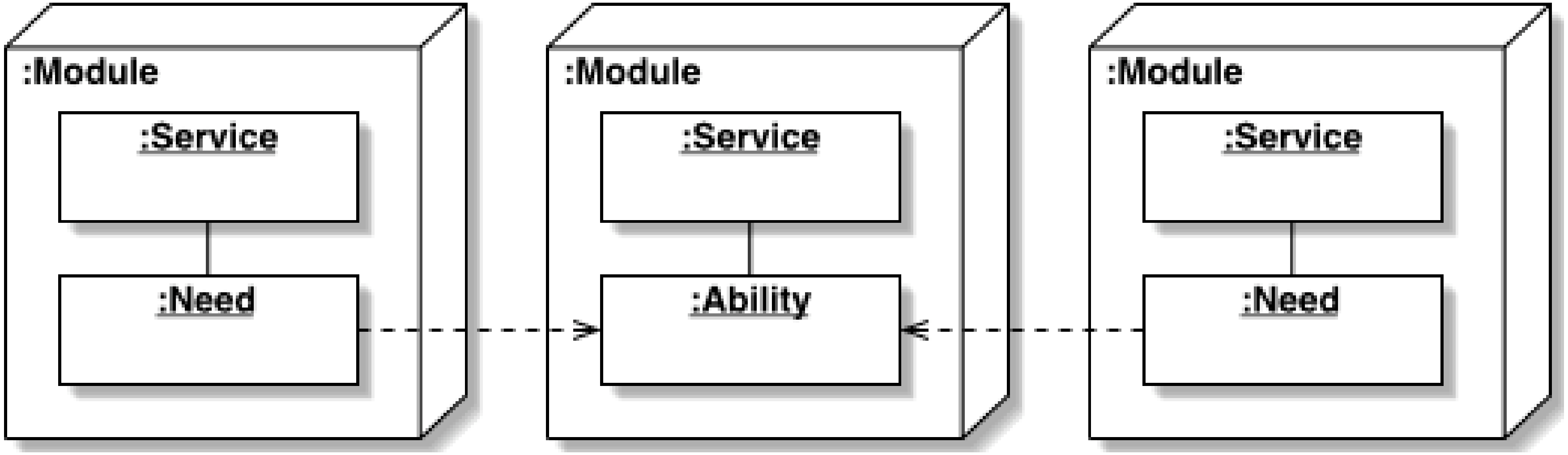
Advantages– Application developer

- System consists of layered *services*
- *Application* is modeled by configuring services



Advantages– Module developer

- Modules are hardware with software services
- Services provide basic functionality, e.g. tracking
- Services have *needs* and *abilities*



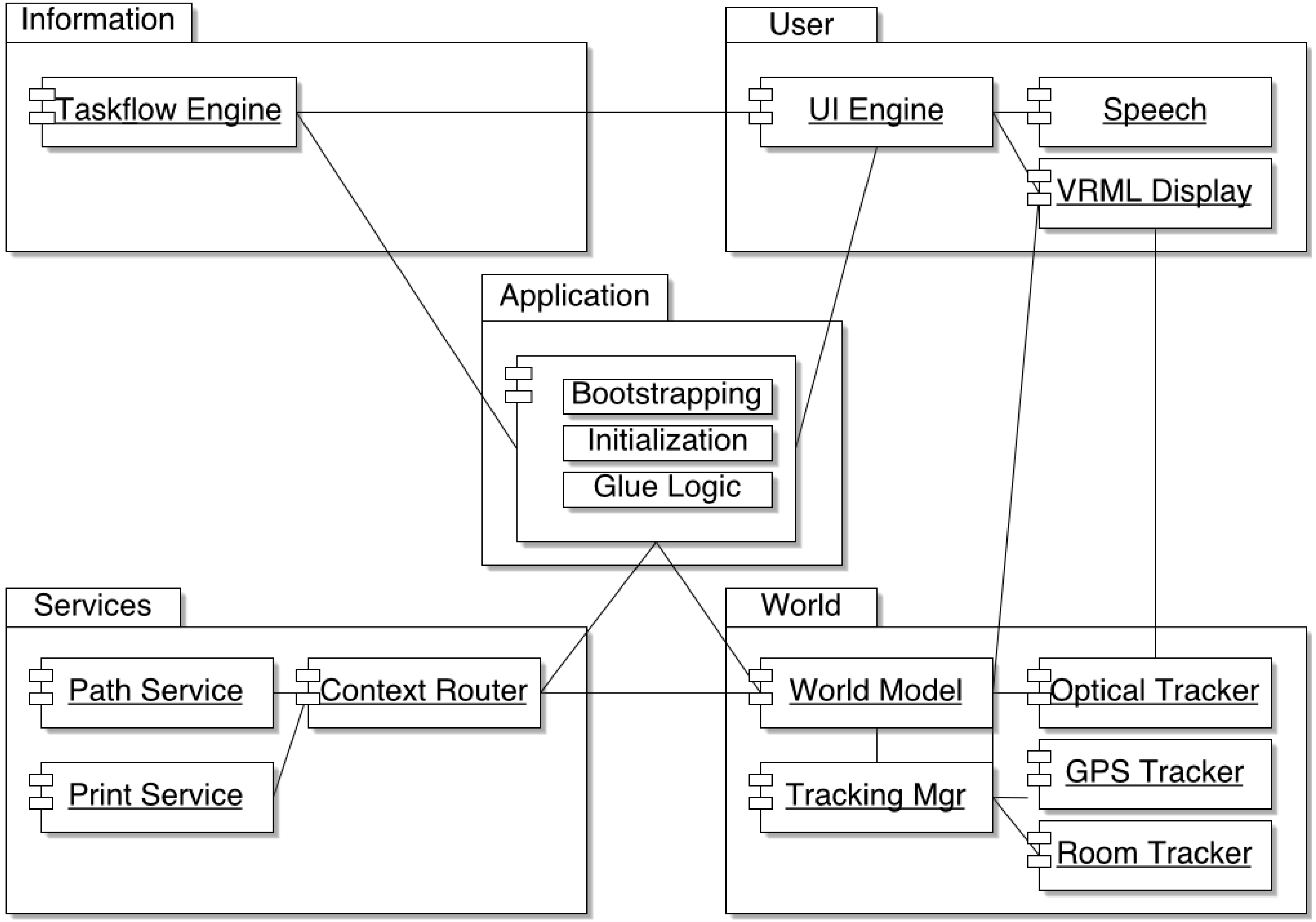
Elements of an AR framework

- Services providing general functionality
  - General enough to reuse, yet also efficient
- Architecture to fit the services together
  - Generalization of many different AR systems
- Middleware to let services communicate
  - Must allow fast yet flexible communication

DWARF– Main services

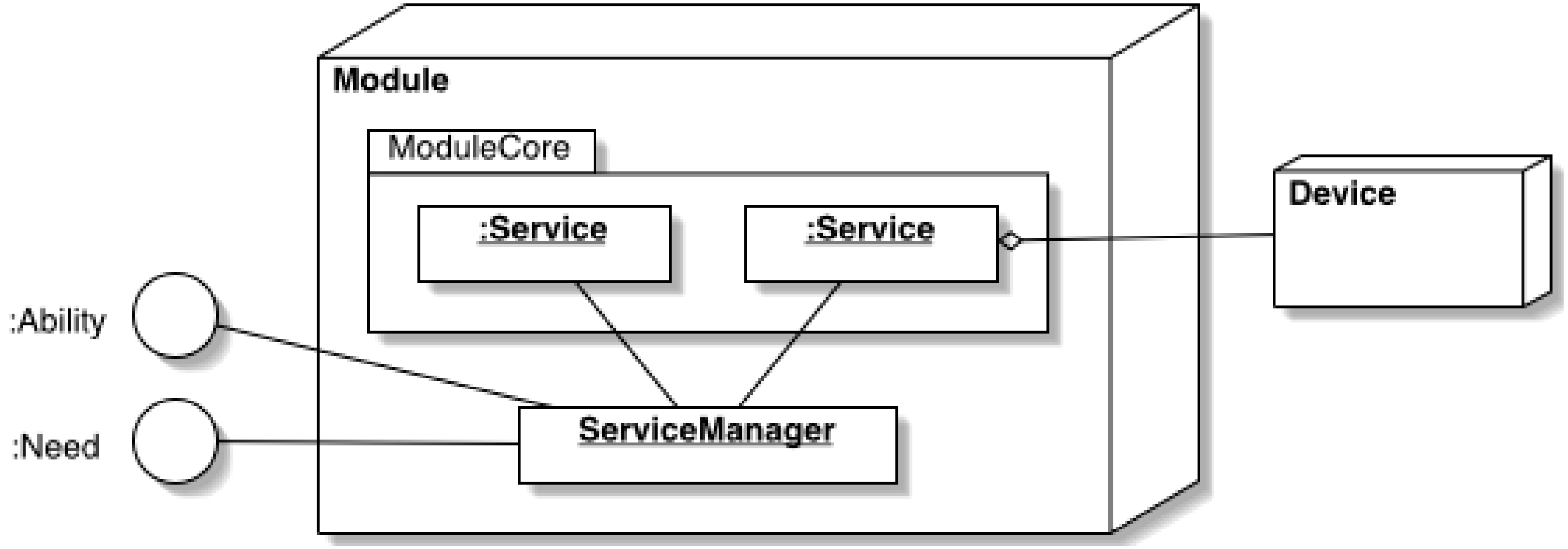
- Distributed Wearable Augmented Reality Framework as test of framework concepts
- First DWARF services cover the basic functionalities for AR applications
  - World model, optical tracker, GPS/compass tracker, tracking manager, user interface engine,VRML viewer, HTML viewer, voice recognition, taskflow engine, context router
- Implementation of services uses well-established and third-party technology

DWARF– Generic architecture



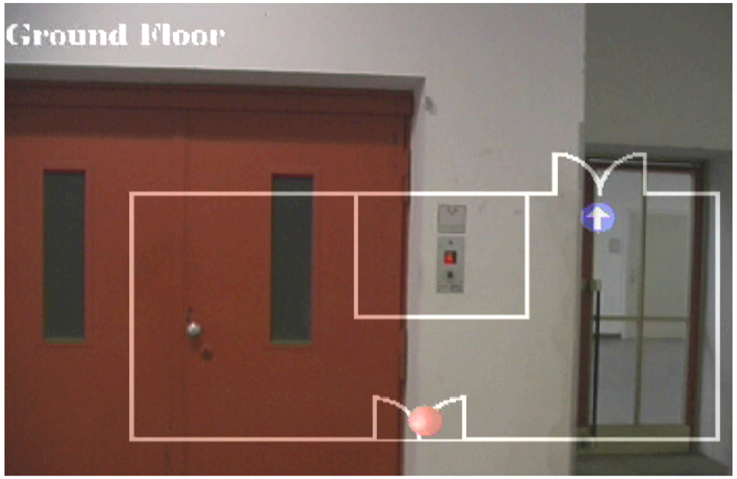
DWARF– Middleware

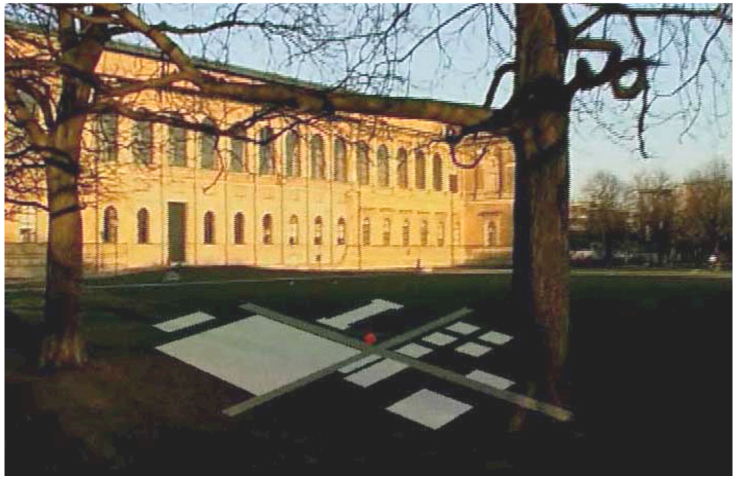
- To find each other spontaneously and communicate with one another, the DWARF services use CORBA-based Middleware
- This is distributed as local Service Managers on each hardware node to provide fault tolerance



Demo system– Navigation scenario

- Indoor and outdoor campus navigation system
- Wireless access to external services
- 2D maps, 3D maps and AR overlays



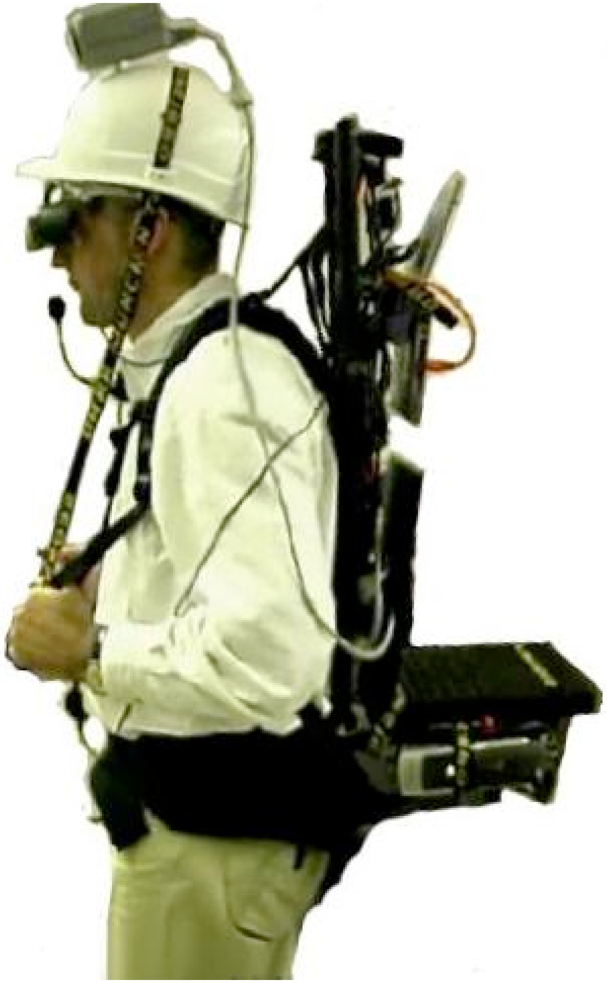


Demo system– Using the framework

- Nearly entire functionality is handled by the DWARF services
  - Navigation sequence, campus geography, multi-modal I/O, printer location, outdoor tracking, indoor tracking, 3D rendering
- Application design is greatly simplified
  - Model campus and navigation in markup languages
  - Bootstrapping, glue logic
- Application implemented in three weeks’ time

Demo system– Hardware

- Backpack system for demonstration of software
- Two laptops running Windows NT and 98
- Glasstron see-though HMD, FireWire Camera for Optical Tracking, commercial GPS receiver, Bluetooth / WaveLAN wireless communication
- Battery-powered, > 2 hours of operation



Conclusion

- The time has come for AR frameworks
  - Technology is becoming mature
  - Number of AR research groups is increasing
- We have investigated the feasibility of AR frameworks
  - Componentization is workable
  - Performance does not suffer from distribution
  - Rapid application development becomes possible
- We would like to encourage a discussion on framework components and technology.

Thank you

- Any questions?