Human Interactions in Dynamic Environments through Mobile Web Services

Daniel Schall, Robert Gombotz, Christoph Dorn, Schahram Dustdar

Distributed Systems Group
Institute of Information Systems
Vienna University of Technology
Austria

www.vitalab.tuwien.ac.at
Talk Overview

- Today’s Collaboration and Challenges
- Problem Statement
- Activity-centric Collaboration
- Human-provided Services
- HpS Application
- HpS Implementation
- Conclusion and Outlook
Today’s Collaboration

- Often Email-based to coordinate activities and exchange artifacts
  - Flexible but hard to manage
- Flexible Team structure
  - Increasingly medium/short lived to work on given task
  - Collaboration across organizational boundaries
- On the move
  - Using mobile devices
  - Spontaneous interactions
  - Situated resources
Challenges

- Humans are key resources in collaboration
  - How to structure collaboration more efficiently?
  - How to describe expertise?
  - How to incorporate human into collaboration?
- Mobile / ad-hoc collaboration
  - How to find available expert to solve problem?
  - How to select right collaboration partner if a number of candidates exist?
  - How to interact with human?
Problem Statement

- Information regarding human (profile, expertise) bound to centralized cooperate directory
  - Difficult to find expert in flexible collaboration scenarios
  - Difficult to establish other metrics for expert finding
- Interactions with human
  - Most tools do not account for spontaneous interactions
  - Support for mobile scenarios/interactions

→ Describe and provide Human Services in collaborative environments
Approach to Human Services

- Representing work through activities
  - Coordinate work among participants
  - Relate to people, tools, services carrying out activity
  - Define templates denoting activity patterns

- Modeling Human-provided Services
  - Describe human skills
  - Definition of interactions with human

- Utilizing context information
  - Physical and logical context sources

- Supporting Human Services
  - Platform and device independent, mobility support
Activities in Collaboration

- We define activities as: A set of one or more related action items performed by a human actor and as perceived by a human actor.

- Example of an activity: an entry in a person’s calendar or to-do list.

- Our goal:
  - Integrate tools and resources (services)
  - Best practices

Activities mentioned in collaboration:
[Hill2006] Beyond predictable workflows: Enhancing productivity in artful business processes, IBM SJ 45-4
Our framework supports:

- XML Schema: defining data model for activities
- Activity Declaration: abstract template
- Activity Instance: representation activities in progress
Example of Activities

1) Browse activity declarations
   • Best practices
   • How-To / guidelines

2) Expand composite activity
   • Atomic activities
   • Composable activities - hierarchical

3) Create activity instance
   • Customizable
   • Add/remove sub-activities

4) Customize sub-activity
   • Assign to team member
   • Assign role

5) Atomic activity
   • Executed by team member
   • Atomic element of complex activity

6) Mapping to services
   • Mapping to a class of services
   • Recommendations / service ranking
Example of Activities (contd.)

- Activity implemented by Human Service
  - Example: “Book Rooms for Attendees” implemented by Human Service
Human-provided Services

- **HpS → Services “implemented” by human actors**
  - Features which cannot be implemented through software only
  - Human actors provide their abilities and work force
- **Example services**
  - Document review service, document translation services
- **Human-provided Services integrated into activities**
  - "Send for review“, ”Get expert opinion“
- **HpS as a conventional service in SOA model**
  - Describe HpS
    - Human characteristics
    - Functional → WSDL
    - Non-functional → Meta information (human profile, skills, …)
  - Find HpS
  - Bind/interact with HpS
Combining Activities & HpS

- Orthogonal cases
- Activity declaration created along with mapping to service class
  - **Service consumer**: executes atomic activity
  - Service lookup and selection → execute HpS
  - HpS mapped to an activity without any concern about how and in which scenario the activity is used
- User-driven integration
  - HpS aware of pending tasks and creates explicitly service mapping
HpsS Interactions

Web Service Provider (HpsS)
- WSDLs
  - WS-Operations:
    - <getActivityAvailability/>
    - <getCurrentLocation/>
    - <getQuote/>
    - <getActivityStatus/>
    - <安排Meeting/>

Collab. Service
- GPS

Map Service

Service Consumer
- Tasks (pending)
- Goals
- Requirements and Constraints

(1) Mapping

Wired Network

(3) Mapping

(4) Register/Deregister Services

(5) Add/Search Activities

Activity Management
- Service Lookup

(6) Service Lookup

Service Registry

(2) Register/Deregister Services

Collaboration Services
- WS_1
- WS_2
- WS_n
Expert finding based on contextual constraints
- Looking for HpS to review document

Service consumer
- Formulate query:
  \((\&\text{category=document.reviews})(|(\text{postalCode=1040})(\text{postalCode=1050}))\text{skill=symbian})\)

Service lookup
- If suitable service available, rank and select
- Otherwise, set activity to pending until fitting service becomes available

Suitable services available
- Person has become available
- Context dependent selection
Hps Application – Service Consumer

1) Search services
• Lookup by service class (e.g. doc review)
• Location-based lookup (specify place → services in proximity)

2) Service tree view
• Service classes
• Specific service instances

Location enhanced service explorer
• Overlay available Hps on map
• GPS coordinates on MapPoint
• Meta information (service class)

Detailed info view
• Service (class)
• GPS (long/lat)
• Semantic location
HPS Application – Service Provider

- **Service provider (HPS)**
- GUI for management
- HPS ad-hoc service registration / discovery
- Manage incoming requests
- Context sensors → GPS traces from external Bluetooth sensor
- Possibly service composition on mobile device → Mashup
  - Compose locally deployed services
  - Create composite service from services on devices nearby (remote applications)
HpS Implementation

- HpS implemented as mobile Web service
  - gSOAP, kSOAP, JSR-172 (consumer only), …
- Deployment in OSGi container
  - OSGi designed for resource constraint devices
  - Platform independent (PCs, mobile phones, sensor platforms)
  - Knopflerfish (~288kB) / Concierge (80kB)
- Discovery
  - Based on jSLP implementation (http://jslp.sourceforge.net/)
  - HpS described by string (ServiceURL)
  - Key/value pairs describe HpS attributes
Related Work


Conclusion and Future Work

- Our framework comprises
  - Activities to structure collaboration
  - Utilizing resources to execute activity
  - Humans expressed as services (HpS)
  - Application of HpS and Implementation

- Future work
  - User profile expressing skills, expertise
  - Ranking of HpS based on social metrics
  - Interactions between HpS & BPEL4People
Thanks for your attention – Questions?

Contact: mailto://d.schall@infosys.tuwien.ac.at
Work: http://www.infosys.tuwien.ac.at/staff/dschall/
Lab: http://www.vitalab.tuwien.ac.at
Project: http://www.in-context.eu