

A Blogging Application for Smart Spaces

Diana Zaiceva, Ivan Galov, Dmitry Korzun

Petrozavodsk State University
Department of Computer Science



9th FRUCT Conference, April 26–29, Petrozavodsk, Russia

Table of Contents

- 1 Smart Blogging
- 2 Ontological model
- 3 Implementation
- 4 Conclusion



Smart Blogging

- **Semantic blogging:** new scenarios of blogging
- **Multi-device:** participation of many devices of different classes
- **Multi-service:** access to several blog-services simultaneously
- **Multi-user proactivity:** simultaneous access to blogosphere, community of users by interests
- **Context-aware scenarios:** recommendation system for blogger, rating of blogs...



Architecture (based on Smart-M3)

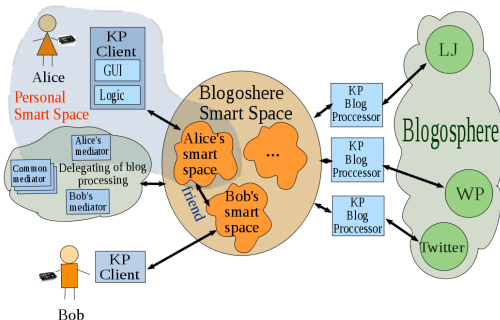
KP Client: content available in the blogosphere, publishing user data

KP Blog Processor: access to information from services

Mediators: aggregation, filtering, processing context data

Blogosphere Smart Space: all blog data in Smart Space

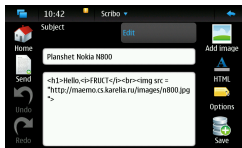
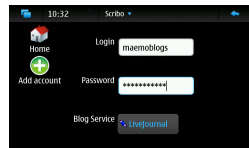
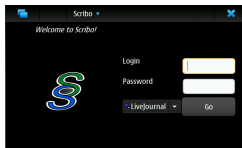
Personal Smart Space: part of Blogosphere Smart Space



Traditional blogging scenarios

- Session initialization and closing
- Account management
- Sending and receiving messages
- Friends management

on the example Scribo client for Maemo 5



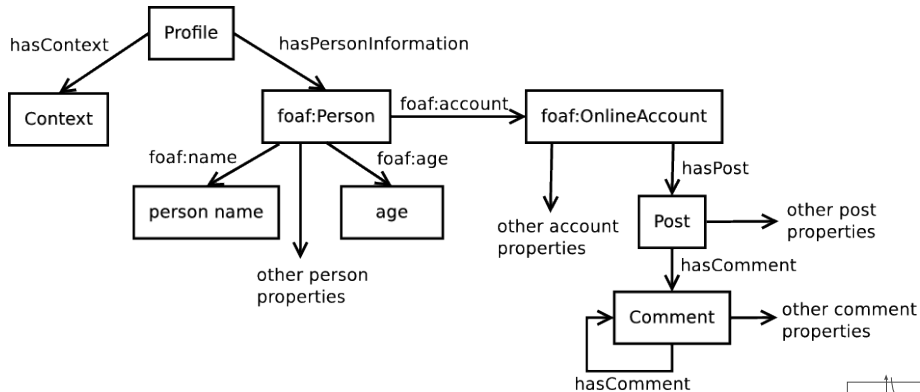
Personal Smart Space

- Person: long-term data
- Context: mutable data
- Blog data: accounts, posts, comments



Blogosphere ontology

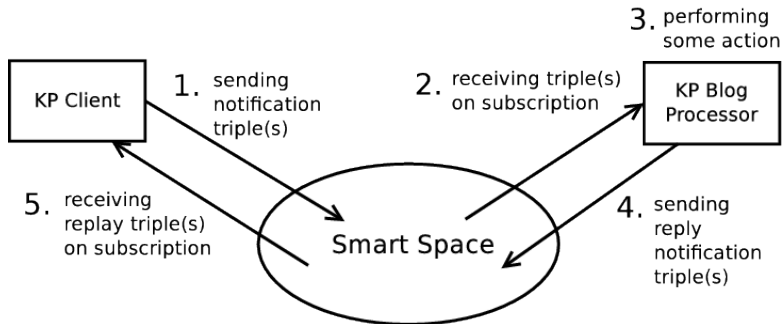
Based on FOAF dictionary



Notification model

Notifications initiate appropriate KPs to execute actions or to inform KPs about the result of execution

Triple: $\langle \textit{subject} \rangle \langle \textit{predicate} \rangle \langle \textit{object} \rangle$

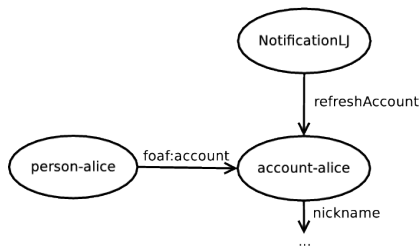


Proactive and reactive notification type



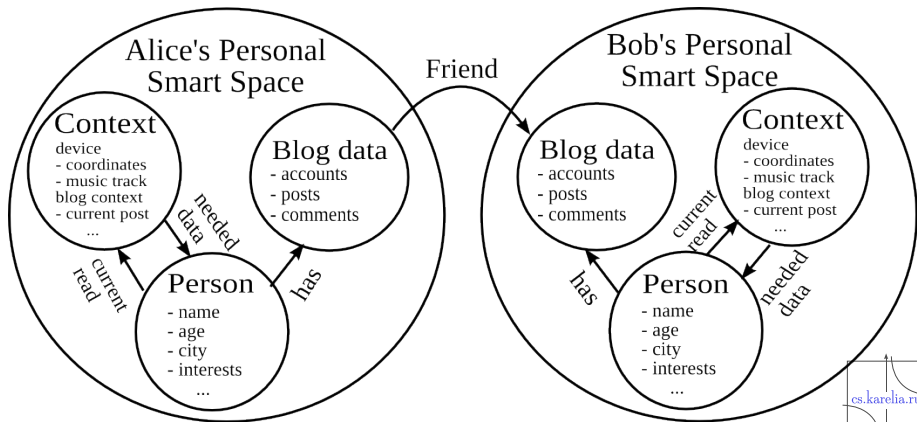
Notification types

- Account notifications
(refreshes user account information)
- Post notifications
(send, edit, delete posts)
- Comment notifications
(send and delete comments)
- Friend notifications
(add, delete, refresh list of friends)



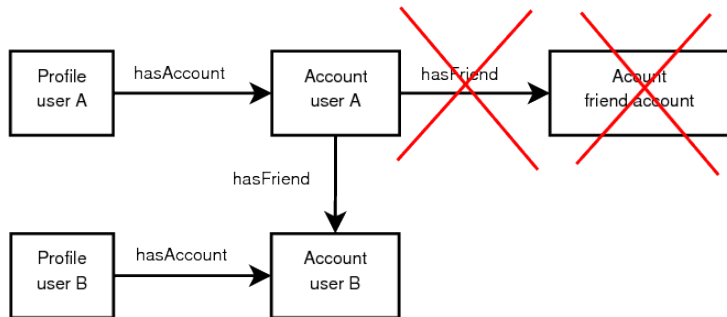
Smart Spaces Composition

Personal smart spaces can logically compose bigger space and interact with each other



Smart scenarios. Delegation and cooperation

- Delegation of processing from client to dedicated servers

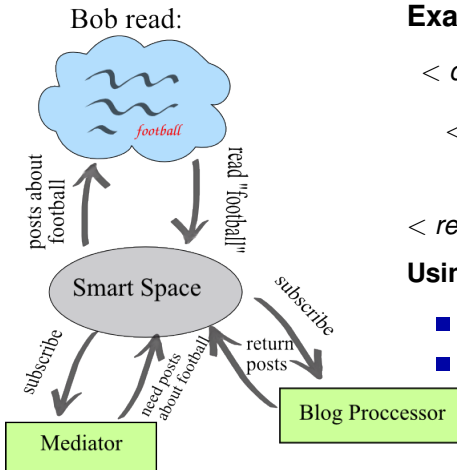


- Cooperation of personal smart spaces

- ▶ friends
- ▶ list of posts of comments
- ▶ interests



Prediction and recommendation



Example:

< context >< place >< Petrozavodsk >

< context >< interest >< tourizm >



< recommend >< post >< post_title_Kizhi >

Using:

- recommendation system
- rating of blogs



Code structure

■ KP Client

- ▶ smartscribo - prototype on PyQt (PetrSU)

■ KP Blog Processor

- ▶ kp_lj - access to LiveJournal blog-service (PetrSU)
- ▶ test_kp_rss - reading data via RSS (PetrSU)
- ▶ kp_lj_ss - integration Blogosphere Smart Space, LiveJournal blog-service and Conference Smart System Smart Conference System (LETI)

■ KP Mediator

- ▶ mediator - not realized

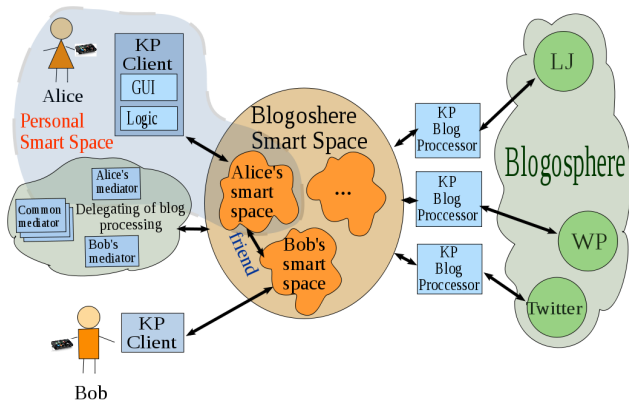
■ Test KP

- ▶ test_client - console client (PetrSU)
- ▶ kp_blog_processor - publishes stub content and handlers notification



Current state:

- sending and viewing posts from LJ
- account management
- comment management (in SIB)
- adding and viewing RSS feed from LJ and Twitter



Results:

- Architecture
- Smart Scribo prototype (traditional scenarios)
- Ontological model:
 - ▶ blogger personal smart space structure
 - ▶ blogosphere ontology based on FOAF
 - ▶ notification model

- Smart Scribo project wiki:

`http://oss.fruct.org/wiki/SmartScribo`

- Sources:

`http://gitorious.org/smart-scribo/smart-scribo`

Please, send your comments to `smart-scribo@cs.karelia.ru`

Thank you for your attention

