

Blogging in the Smart Conference System

Dmitry Korzun, Ivan Galov, Alexey Kashevnik, Nikolay Shilov,
Kirill Krinkin, Yury Korolev

Petrozavodsk State University
Department of Computer Science

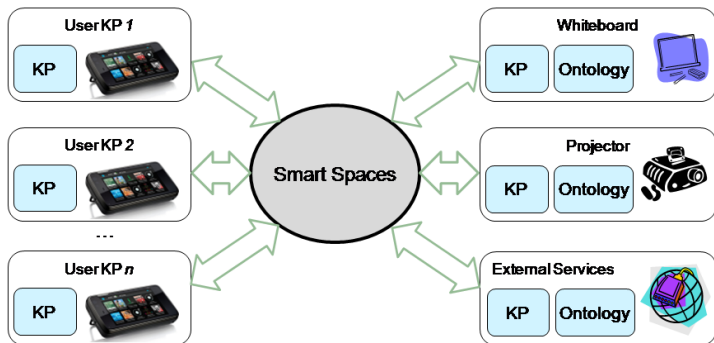


9th FRUCT Conference, April 26–29, Petrozavodsk, Russia

Table of Contents

- 1 Integration of Smart Applications: reference use case
- 2 Ontology representations
- 3 SC Blog Processor
- 4 Conclusion

Smart Conference System

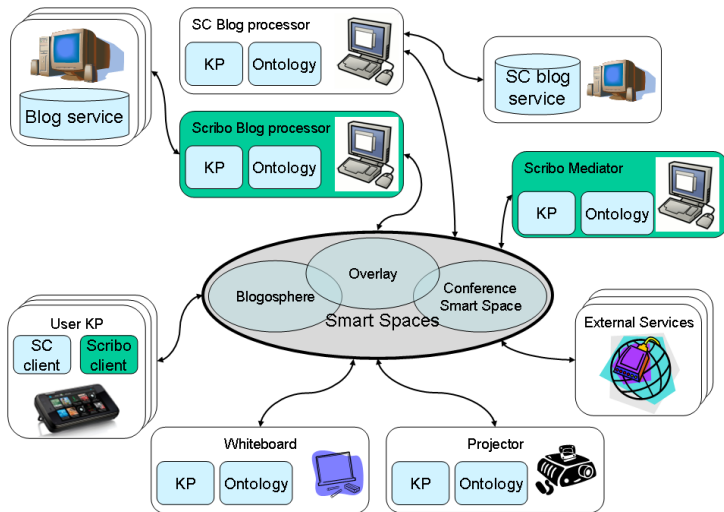


Topic-based discussions

Possibility to discuss the conference is needed:

- questions for speaker
- speaker's answers
- discussing current or previous presentations
- debate with participants

SmartScribo system for blogging



Blogging for conference

- each conference – a separate blog
- one post per a talk/paper
- posts for other discussion threads
- comments – participants' discussions

post

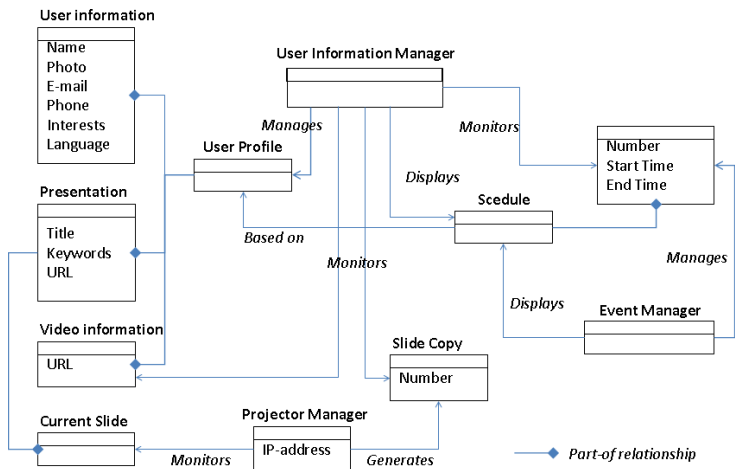
- comment1

...

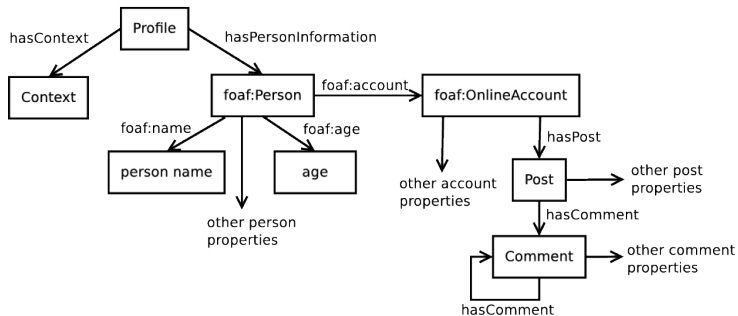
- commentN

- ▶ comment to comment

Conference Ontology (in SCS)

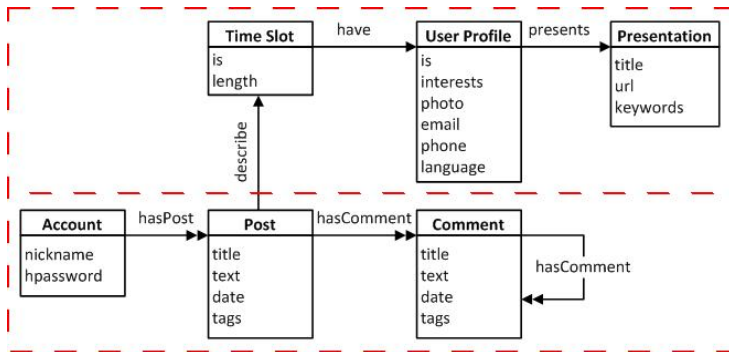


Blogosphere Ontology (in SmartScribo)

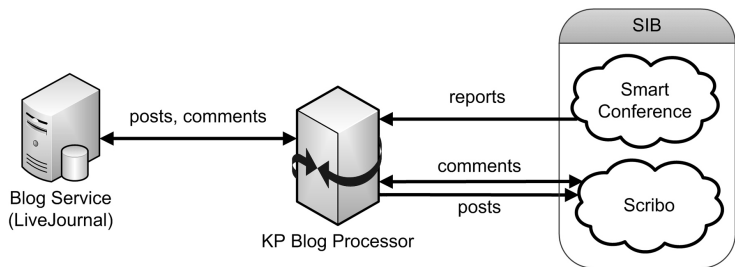


Overlay ontology

connects fragments from both ontologies



SC Blog Processor data flows



Notifications:

- refreshPosts – loading reports
- refreshComments – comments synchronization
- sendComment – comments publishing

SC Blog Processor functions

Data flow	Interaction with LiveJournal	Interaction with Smart Space		Synchronization of different spaces
Module	LiveJournalHandler Read / Write	ConferenceHandler Read	ScriboHandler Read / Write	Synchronizer Local actions
Basic functions	<ol style="list-style-type: none"> 1. Receiving posts. 2. Sending post. 3. Receiving comments hierarchy of a post. 4. Sending comment. 	<ol style="list-style-type: none"> 1. Loading report descriptions 2. Tracking changes in reports 3. Tracking changes in the schedule 	<ol style="list-style-type: none"> 1. Loading information about comments 2. Tracking new comments from blog clients. 3. Tracking requests for data from blog clients 4. Sending notifications to blog clients 	<ol style="list-style-type: none"> 1. Synchronization 2. Local storage of posts (reports) and comments based on common GUID 3. Store synchronized GUID map for items in different spaces

Conclusion

- SC system: SPIIRAS

<http://sourceforge.net/projects/smartconference/>

- SmartScribo: PetrSU

<https://gitorious.org/smart-scribo>

- SC Blog Processor: SPbETU

<https://github.com/kua/scblog>

Results:

- architecture of the integration
- ontological model
- integration scheme

Thank you for your attention