CNS: A Content-centric Notification Service

Jiachen Chen[1], Mayutan Arumaithurai[1], Xiaoming Fu[1], K.K.Ramakrishnan[2]
[1] *University of Goettingen, Germany;* [2] *AT&T Labs-Research, U.S.A.*

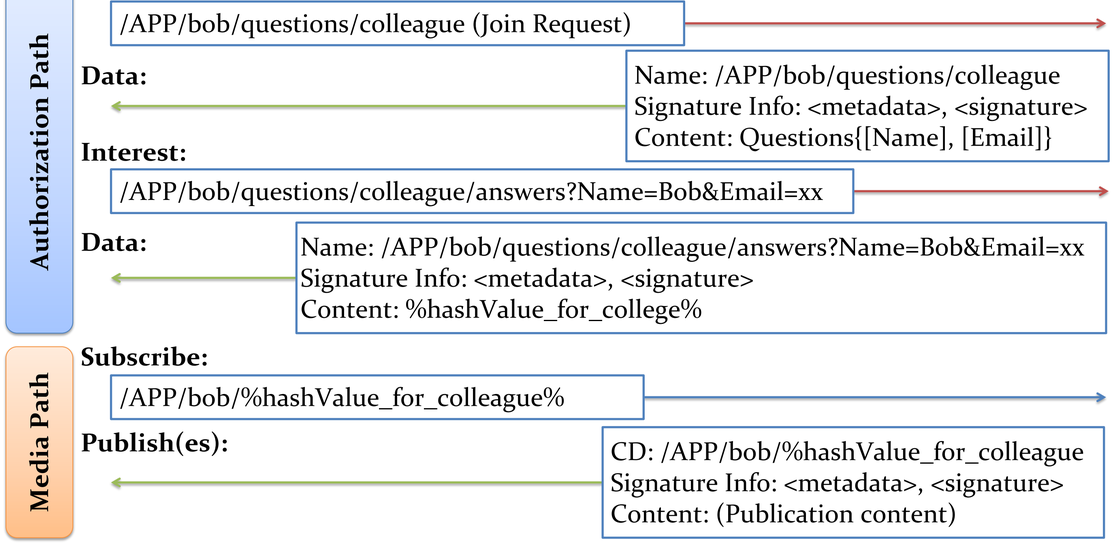
Motivation:

- Need for a ***scalable***, ***secure*** and ***timely*** notification system:
  - Publish data to certain groups of people, e.g., close friends, office-colleagues, etc.
- Useful in Disaster Management, pub/sub environments
- Twitter like pub/sub applications:
  - Useful but not scalable
  - Place load on servers, rely on frequent polling, closed system

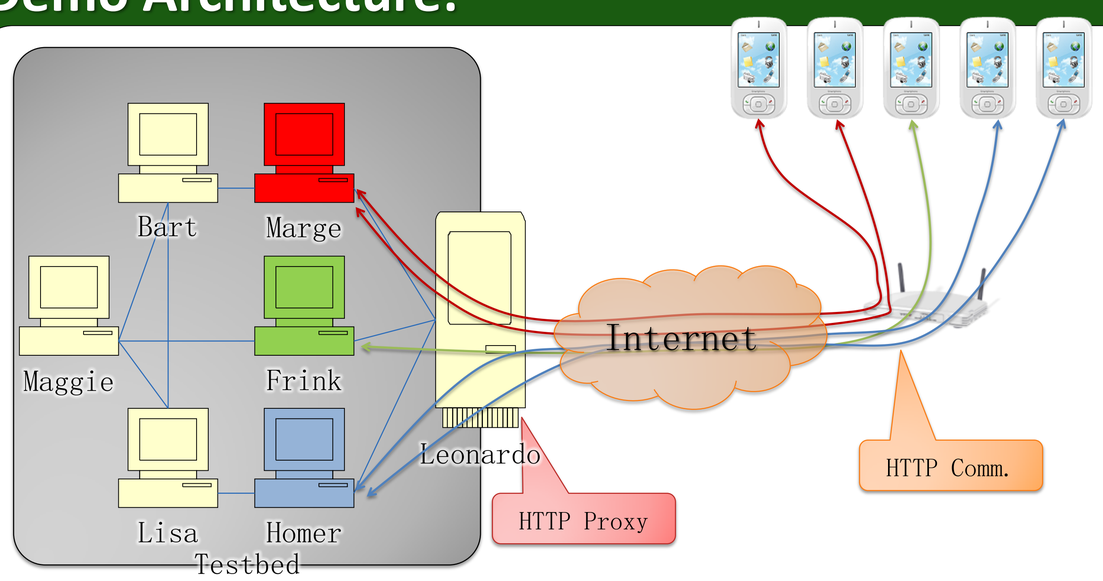
Content-centric Notification System (CNS):

- Content-centric [1,2,3] Notification System
- Hierarchical group management
- Based on COPSS [3]
  - Supports highly dynamic and scalable pub/sub systems
  - Hierarchical content descriptors
  - Push based multicast support
  - Publishers need not have apriori info about subscribers
- Timely notification from users to subscribers
- Effective authorization (allows Third party authorization too)

Protocol Exchange for Group Joining:



Demo Architecture:



Client GUI Demonstration:

User Profile Management

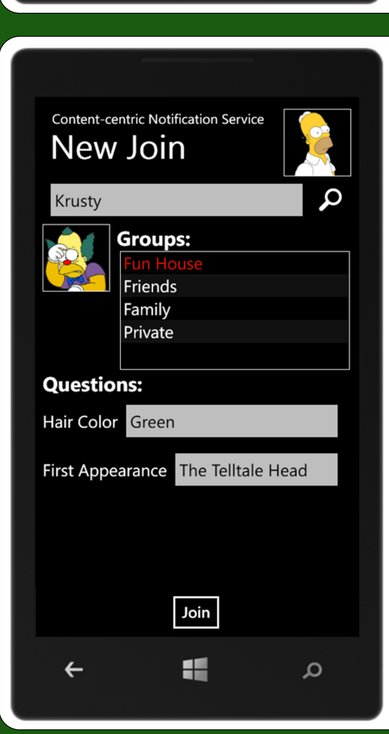
- When the app is launched for the first time, a user needs to initiate his profile. User can always come back and edit this page (When the user leaves this page, the 'profile' icon will show up).
- Click picture to update profile picture.
- Add a field by typing property and value at bottom.
- Click a field to edit the selected field.
- Remove a field using the Delete button at bottom right.

Content-centric Notification Service - Application Design 2

Own Groups Management

- When the user profile is set, the app directs the user to the Own Group Management page.
- Alternatively, he can edit his own groups by clicking on the 'owned group' icon.
- Own Group Management shows different groups that are owned by the user and the fields needed for authorization in that group.
- Click a group to edit or click "New" to add a new group.
- Add/remove fields by clicking the checkbox at bottom.
- Finish editing by clicking the Tick button.
- Delete a group by clicking the Delete button.

Content-centric Notification Service - Application Design 3



Join a New Group

- The user fills the user name in the search field.
- [The App requests for the available groups & portrait].
- The user selects a group to join.
- [The App requests for the authorization fields].
- The user answers the questions and click the "Join" button.
- [The App requests for the group ID].
- [If the answer is correct, the App will join the group].
- [If the answer is wrong, the App notifies the user].
- The user can then search for another user again or click another group to follow.
- The user can go back to Joined Group Management page by clicking the hardware "back" button.

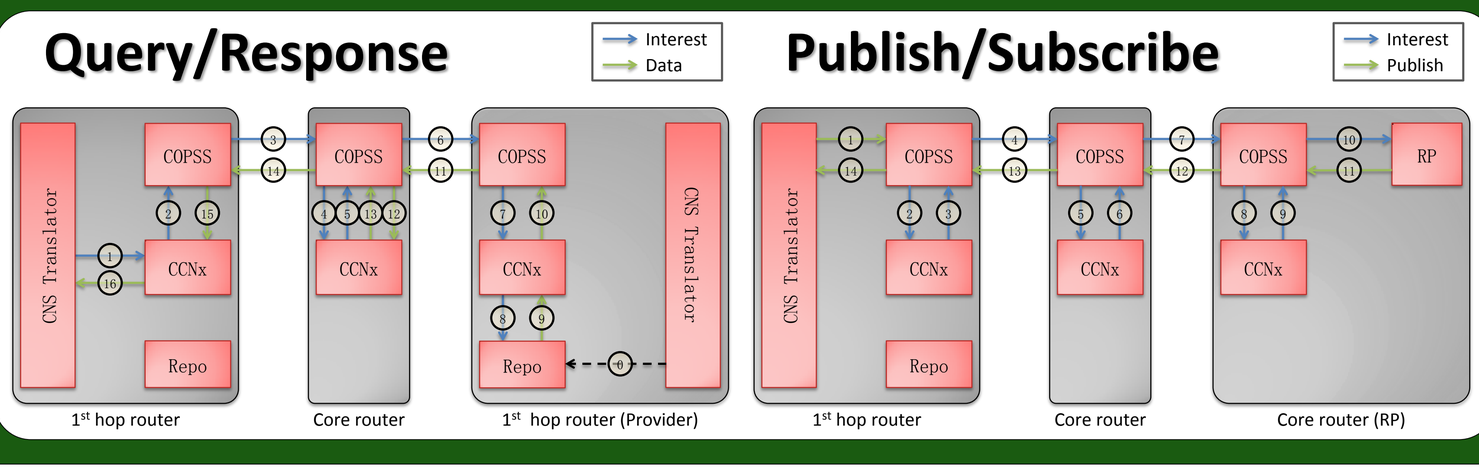
Content-centric Notification Service - Application Design 5

Conclusion:

This work demonstrates:

- The benefits of COPSS for a notification system
- Hierarchical group management
- A first-step authorization

Data Flow in CNS:



References:

1. V. Jacobson, D. K. Smetters *et al.*, "Networking Named Content," in CoNEXT, 2009.
2. L. Zhang, D. Estrin *et al.*, "Named Data Networking (NDN) Project," PARC, Tech. Report NDN-0001, 2010.
3. J. Chen, M. Arumaithurai *et al.*, "COPSS: An Efficient Content Oriented Pub/Sub System," in ANCS, 2011.

Acknowledgment:

The work for this paper was performed in the context of the FP7/NICT EU-JAPAN GreenICN project.