Service Subscription for Personal Web

Chunyang Ye
University of Toronto
Toronto, Canada
chunyang.ye@utoronto.ca

Hans-Arno Jacobsen
University of Toronto
Toronto, Canada
jacobsen@eecg.toronto.edu

1. INTRODUCTION

Niklaus Wirth ever wrote down in his book a famous philosophy about computer programs, that is, *Algorithms + Data Structures = Programs* [7]. To date, the representation of data structures has evolved from functional languages, object-oriented programming, relational database, semi-structured data(XML) etc to *Semantic Web* [5]. Semantic Web is a web of data. By representing information as linked data and making them referable, Semantic Web provides a unique interface to access heterogeneous and distributed data sources from all over the world.

On the other hand, the implementation of algorithms has also been shifted from basic building blocks in functional languages to more well encapsulated object-oriented classes and components. Nowadays, the service-oriented architecture (SOA) is a new software engineering paradigm to design, implement and publish reusable software components as Web services [3]. By composing together suitable Web services from third-parties, service consumers can quickly develop distributed Internet applications across different organizations.

In this position paper, we extend Wirth's philosophy with a new interpretation about Internet applications based on the combination of Semantic Web and SOA technologies, that is, *Services + Semantic Web = Internet Applications*. Different from traditional programs where data structures and algorithms are built local to programs, the Semantic Web and SOA technologies allow the data and services to be shared among customers through standard interfaces (e.g., RDF [4] and WSDL [6]), respectively. Therefore, users (customers) are allowed to customize their own applications by consuming well selected services and data over the Internet.

This new vision enriches the development and usage patterns for Web applications. In particular, with respect to Personal Web applications, end-users (customers) can publish their personal information (e.g., comments on a movie, online shopping of particular products etc) as linked data on their personal web and share the data to their friends (e.g., the social network in Facebook [1]). By subscribing to the update of linked data of interest, end-users can get notified of their friends' interesting activities. On the other hand, end-users may not be restricted to get the notification of linked data update only. They may subscribe to some services to manipulate the updated linked data when the data is ready. For example, users may subscribe to the recommended movie information from their friends in the social network website. They may also subscribe to the online box office service to book the tickets based on their online calendar data. When a movie is recommended, the subscribed online box office service will be invoked to book the tickets for the users automatically.

The challenges of this new paradigm for Personal Web applications lie in how to glue the Semantic Web and SOA technologies to develop customized applications for customers. On the one hand, since the Semantic Web and SOA adopt different standards and protocols to share and exchange data, the interoperability between both technologies is a challenge. On the other hand, the data and services usually belong to different organizations that are beyond the control of customers, how to coordinate the data and services is another challenge.

In the rest of this paper, we present our solution to address the aforementioned challenges.

2. OVERVIEW OF METHODOLOGY

As mentioned in Section 1, the challenges of our proposal lie in the interoperability between Semantic Web and SOA technologies and the coordination of data and services that are not under the control of customers. In this section, we briefly introduce our solutions to address the challenges.

In order to interoperate between the Semantic Web and SOA, we extend our previous work [8] to allow services access the data from the Semantic Web based on event interfaces. An event interface declares what kinds of events a service wants to expose and what kinds of events it wants to subscribe. An event is defined as a state change. For example, if some linked data in the Semantic Web is updated, an event can be generated and propagated to the services that use the linked data. On being notified of the subscribed events, the services can update the copies of the linked data inside the services. On the other hand, if a service updates the copies of some linked data, then events will be raised and exposed. The exposed events are then used to update the corresponding linked data in the Semantic Web. Therefore, as illustrated in Fig. 1, we provide our Polaris framework [9] to bridge the gap between the linked data in Semantic Web and SOA services. By declaring the state changes related to

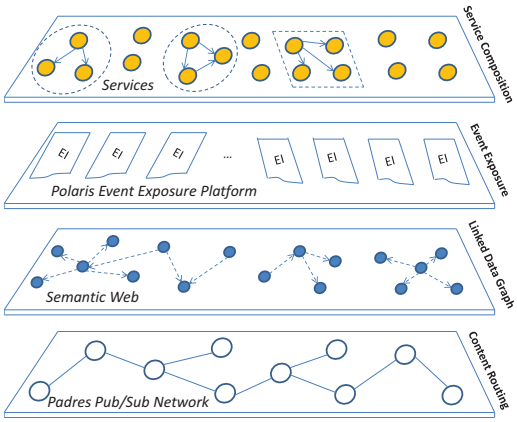


Figure 1: Overview of methodology.

the linked data of interest, services can access and synchronize with the linked data.

Another issue is to invoke the subscribed services when the subscribed linked data is ready. To do so, the Polaris event exposure platform allows users to design some Event-Condition-Action(ECA) rules. When an event is notified, the corresponding matched ECA rules are triggered to invoke the subscribed services transparently and asynchronously.

Since the linked data is distributed over the Web, an event related to the changes of linked data needs to propagated from the event producer to the event subscribers. Therefore, we use padres [2], a content-based pub/sub middleware to propagate events. Padres allows users to advertise and subscribe events based on their contents. When an event is published, it will be automatically delivered to all the subscribers whose subscriptions match the content of the event. With respect to Personal Web applications, users can subscribe to linked data from their friends by specifying the subscription with the social network information. As illustrated in Fig. 1, the Semantic Web built on top of the padres pub/sub network is allowed to route the changes of linked data to the subscribers transparently with social network information.

Therefore, by adopting the Padres pub/sub network and Polaris event exposure platforms to bridge the gaps between the Semantic Web and SOA services, our solution allow personal web users to customize their applications based on subscribed Semantic Web data and SOA services.

3. FUTURE WORK

The current work presented in this paper provides some preliminary functionality to prove the concept of service subscription for Personal Web applications based on our Polaris and Padres framework. In the future, we plan to complement the current work with the following features:

- **Service Subscription Language**. In our current work, users specify which services are invoked to handle the subscribed linked data using the ECA rules. This imposes difficulty for end-users to design customized applications. We plan to provide a service subscription language for end users to easily specify what services they need. The service subscription language should also be able to specify some non-functional requirements (e.g., a user may want to subscribe the services used by their friends). In this way, users do not have to find the services and link them to the Semantic Web data by themselves.

- **Service Discovery and Matching**. With given service subscription, we plan to develop solutions to discover and match the subscribed services. Different from the subscription of linked data, the service subscription may include the behavior description of services. This requires to extend traditional content-based matching algorithm to support behavioral matching. In addition, the matching algorithm may also need to rank and filter the services based on the social network information (e.g., candidate services recommended by friends are marked with higher priority).

- **Service Wrapper**. In order to make use of the linked data, services need to provide an event interface to describe how the data internal to a service and the linked data outside the service are exchanged. This task however is tedious and error-prone. Therefore, in our future work, we plan to provide a tool for services to generate wrappers automatically to map the data between services and the Semantic Web with event interfaces.

- **Linked Data Advertisement and Subscription**. In the Semantic Web, data from different locations is linked. The linking relationship between data forms linked data graphs. In some applications, if one data is changed, some other data linked to the changed data may also be needed. For example, if a user updates the linked data about his/her favorite movies, the information about the new movies are usually also needed by users. Therefore, we plan to explore the linked data graph to generate the related advertisements and subscriptions automatically. In this way, users can get the notification of related data transparently.

4. REFERENCES

[1] Facebook. http://www.facebook.com/.
[2] G. Li, V. Muthusamy, and H.-A. Jacobsen. A distributed service-oriented architecture for business process execution. *ACM Trans. Web*, 4(1):1–33, 2010.
[3] M. P. Papazoglou, P. Traverso, I. Ricerca, and S. Tecnologica. Service-oriented computing: State of the art and research challenges. *IEEE Computer*, 40:2007, 2007.
[4] W3C. Resource Description Framework. http://www.w3.org/RDF/.
[5] W3C. Semantic web. http://www.w3.org/RDF/FAQ.
[6] W3C. Web Service Description Language. http://www.w3.org/TR/wsdl.
[7] N. Wirth. *Algorithms + Data Structures = Programs*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1978.
[8] C. Ye and A. Jacobsen. Event exposure for web services: A grey-box approach to compose and evolve web services. pages 1–21. Springer-Verlag, LNCS Volume 6400, 2010.
[9] C. Ye and A. Jacobsen. Polaris: a framework to compose and evolve smart web services via event exposure. In *IBM CASCON Exhibits*, 2010.

Service Subscription for Personal Web

Chunyang Ye and Hans-Arno Jacobsen
University of Toronto
2010

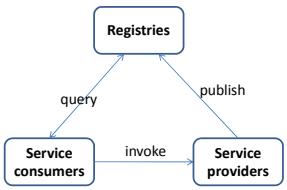
Evolution of Computer Programs

Algorithms + Data Structures = Programs*

- Algorithms
  - Functional blocks
  - Object-oriented classes
  - Software components
  - Web Services
- Data Structures
  - Functional language
  - Object-oriented classes
  - Relation database
  - Semi-structure (XML)
  - Semantic Web
- Programs*
  - Personal Web Applications

*N. Wirth. Algorithms + Data Structures = Programs. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1978

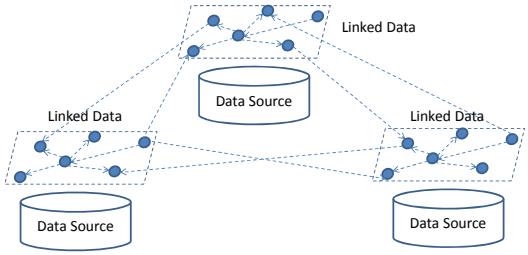
Service-oriented Architecture



1) Service providers develop reusable software components and publish them as Web services in the public registries.

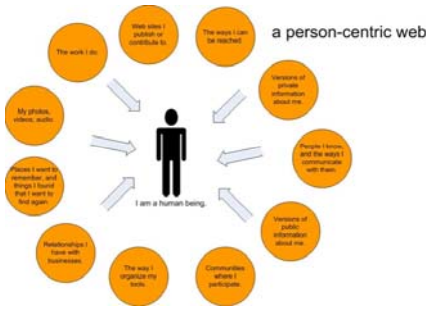
2) Service consumers query public registries, select suitable services and compose them to develop their own applications.

Semantic Web



Semantic Web is a web of data, providing a unified interface to access data from different data sources over the Internet.

Personal Web



* This figure is from http://marynabadenhorst.globalteacher.org.au/pln/

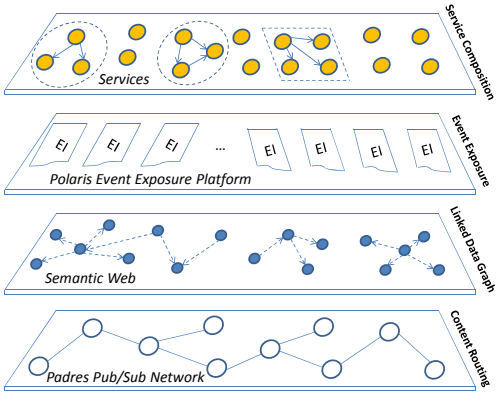
5

Service Subscription

- How to customize services for Personal Web?
  - E.g., Subscribe to comments on the movies friends are interested in.
  - E.g., Subscribe to the online box office service to book the tickets based on the recommendations from friends.
- Challenges
  - How to combine Semantic Web and SOA?
    - Since they are beyond the control of users.

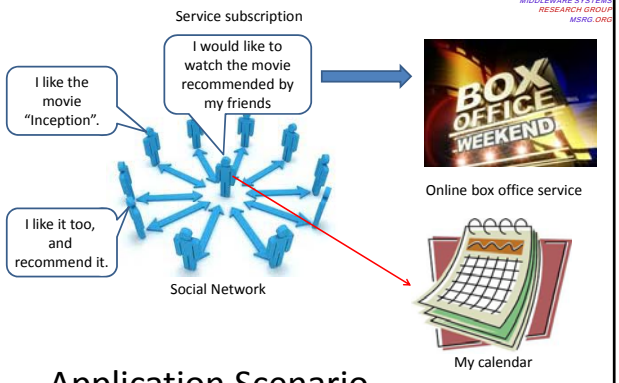
6

Overview of Methodology

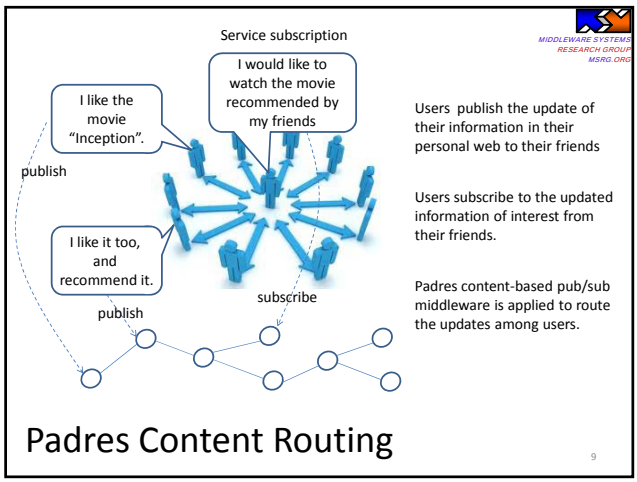


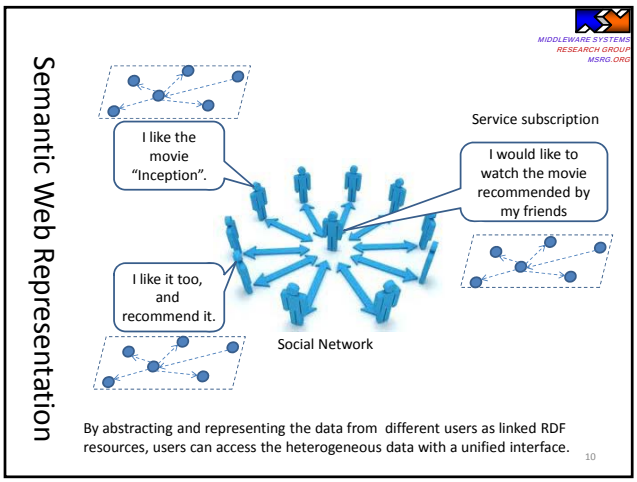
7

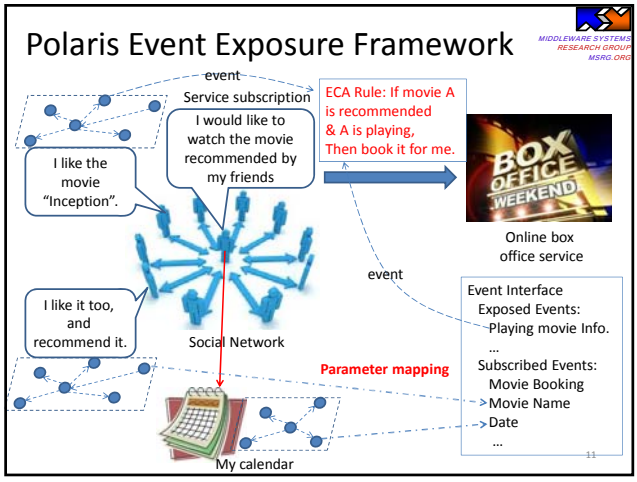
Application Scenario

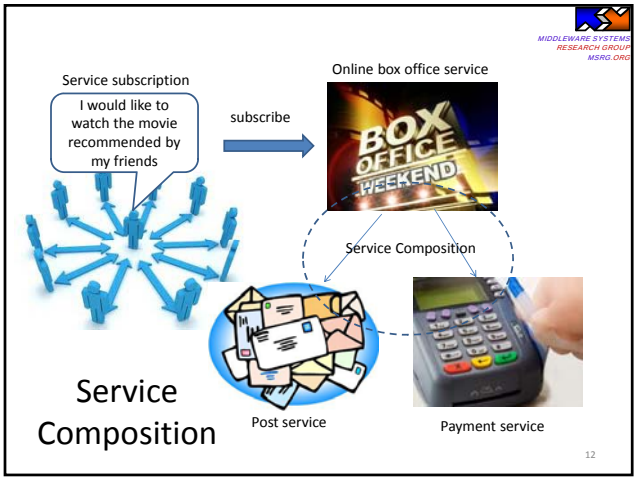


8


Service subscription
I would like to watch the movie recommended by my friends
I like the movie "Inception".
publish
I like it too, and recommend it.
publish
subscribe
Users publish the update of their information in their personal web to their friends
Users subscribe to the updated information of interest from their friends.
Padres content-based pub/sub middleware is applied to route the updates among users.
Padres Content Routing
9


Semantic Web Representation
Service subscription
I would like to watch the movie recommended by my friends
I like the movie "Inception".
I like it too, and recommend it.
Social Network
By abstracting and representing the data from different users as linked RDF resources, users can access the heterogeneous data with a unified interface.
10


Polaris Event Exposure Framework
event
Service subscription
I would like to watch the movie recommended by my friends
I like the movie "Inception".
I like it too, and recommend it.
Social Network
ECA Rule: If movie A is recommended & A is playing, Then book it for me.
Online box office service
event
Event Interface
Exposed Events:
Playing movie Info.
...
Subscribed Events:
Movie Booking
Movie Name
Date
...
Parameter mapping
My calendar
11


Service subscription
I would like to watch the movie recommended by my friends
subscribe
Online box office service
Service Composition
Service Composition
Post service
Payment service
12

Future Work

- Service Subscription Language
  - Describe what services are needed
  - Behavioral perspective
- Service Discovery and Matching
  - Advertisement and Behavior Matching
- Service Wrapper
  - Automatically generate the event Interfaces to map the linked data and services' local data
- Linked Data Advertisement and Subscription
  - Correlation of linked data

13