The Nexus of SOA, Cloud and Crowd

Hailong SUN
sunhl@act.buaa.edu.cn
Beihang University
2013/10/23
Outline

- Background
- SOA, Cloud and Crowd
- Our recent work

Gartner 2012: The Nexus of Four Forces

- Cloud computing
- Mobile Internet
- Social network
- Information (Big Data)

What about SOFTWARE?
(1) Ubiquitous services

- **End-users:** BYOD—Bring Your Own Device
  - Web APP
  - Mobile APP
- **Development and Maintenance**
  - Dev: Web services, Web API
  - Runtime: PaaS, IaaS

**Fast growing of Web APIs**

<table>
<thead>
<tr>
<th>Year</th>
<th>API Growth</th>
</tr>
</thead>
<tbody>
<tr>
<td>2005/8</td>
<td>40m</td>
</tr>
<tr>
<td>2008/11</td>
<td>31m</td>
</tr>
<tr>
<td>2011/5</td>
<td>3m</td>
</tr>
<tr>
<td>2011/10</td>
<td>3m</td>
</tr>
<tr>
<td>2012/2</td>
<td>3m</td>
</tr>
<tr>
<td>2012/5</td>
<td>4m</td>
</tr>
<tr>
<td>2012/8</td>
<td>5m</td>
</tr>
<tr>
<td>2013/4</td>
<td>7K</td>
</tr>
<tr>
<td>2013/11</td>
<td>8K</td>
</tr>
<tr>
<td>2014/3/16</td>
<td>9K</td>
</tr>
</tbody>
</table>
Large-scale use of Web APIs

- 13 billion API calls / day (May 2011)
- 10 billion API calls / month (January 2011)
- Over 260 billion objects stored in S3 (January 2011)
- 1.6 billion API-delivered stories / month (October 2010)
- 5 billion API calls / day (April 2010)
- 5 billion API calls / day (October 2009)
- 8 billion API calls / month (Q3 2009)
- 3 billion API calls / month (March 2009)

Courtesy to John Musser, ProgrammableWeb

Large-scale use of Web APIs (cont’)

- 1.1 billion API calls / day (April 2011)
- 12 billion API calls / month (May 2011)
- 5 billion API calls / month (May 2011)
- 2 billion API calls / month (September 2010)
- 1.5 billion API calls / month (May 2011)
- 1 billion API calls / month (September 2011)

Offline, closed software→Online, open services

Courtesy to John Musser, ProgrammableWeb
(2) Being Networked: complicated connections

- **People**: Social Network
  - User
  - Developers
  - Providers
- **Component service**: Dependency
  - Cooperation
  - Similarity
- **Runtime**: Internet
  - Mobile Internet

**Software-Reliant Systems: What HAS Changed?**

**Increased connectivity**

Challenges
- scale and complexity
- decentralization and distribution
- "big data"
- increased operational tempo
- mismatched ecosystem temps
- vulnerability
- collective action
- disruptive and emerging technologies

(3) User participation: crowdsourcing

- Data problem (hard for computers)
  - “Bad Data” (Data Quality): inaccurate, incomplete, duplicate, stale.
  - Unstructured data: text processing (translation, edit), multimedia processing (labeling, classification and object identification)
  - Subjective evaluation: sentiment analysis, ranking, rating
- Mobile sensing (Participatory sensing)
- Web 2.0, UGC and Q&A are all Crowdsourcing!”

What is Crowdsourcing?

Hard for computers!

Requestors

The Task

Task splitting

Platform

Aggregation

Task assignment

Workers (HPU)
Wisdom of Crowds (WoC)

Requires

- Diversity
- Independence
- Decentralization
- Aggregation

Input: large, diverse sample
(to increase likelihood of overall pool quality)

Output: consensus or selection (aggregation)

More about crowdsourcing

Increasing interests from academia

Crowdsourcing ≠ outsourcing

• OS: business model, the contracting out of a business process to a third-party, for saving costs.

• CS: computing model, targeting at an undefined public. *Crowdsourcing is an online, distributed problem-solving and production model* (Daren C. Brabham).

Crowdsourcing → Crowd Computing

Processing Unit: CPU → CPU + HPU
Gartner Hype Cycles for Emerging Technologies 2013

Technology focus: relationship between Human and Machine.

53 years ago

Man-Computer Symbiosis

J. C. R. Licklider
IRE Transactions on Human Factors in Electronics, volume HFE-1, pages 4-11, March 1960

“The hope is that, in not too many years, human brains and computing machines will be coupled together very tightly, and that the resulting partnership will think as no human brain has ever thought and process data in a way not approached by the information-handling machines we know today.”
Theoretical aspects

• Five Deep Questions in Computing.
  – P=NP
  – What is computing? (Given that humans and machines have complementary computing capability, now ask: what is computable?)
  – What is intelligence?
  – What is information?
  – (How) can we build complex systems simply?

From the perspective of SE: crowd participation

• **Opensource**: OW2, Apache, Eclipse, Linux
• Development of mobile APPs
• Mashup: ProgrammableWeb
• Software community: stackoverflow
• Software competition: Topcoder
• Software testing: A/B test, uTest, Baidu test
• Software rating: AppStore
• Software analysis: Windows error reporting
Crowdsourcing in industry

• Google: the world biggest secret crowdsourcing platform
• Oracle: integrating Crowdsourcing with CRM software to collect customer requirement
• Microsoft: leverage crowdsourcing to improve the security of Windows 8.

• In China, Xiaomi mobile phone improves MIUI recording software based on crowdsourcing.
• Baidu test leverages user contribution to test various services and Apps.
End user programming

There are lots of end users

Using data from the Bureau of Labor Statistics, we estimate that over 90M Americans will use computers at work in 2012. Of these, only about 2.5M will be professional programmers; 40.5M will be managers and (non-software) professionals.

This does not include home users or non-US users, so there will be many more than 90M total end users. Most of them will “program” in some way.

Courtesy to Mary Shaw, CMU.

Internet users

http://www.internetworldstats.com/stats.htm
The role of software engineering

Separation of users from software processes!

Social-Technical Trends

IBM, Clay Williams: The social side of software engineering
Summary

- **Ubiquitous services**
  - Beyond SOAP services
  - Web API, APPs

- **Being networked**
  - Building services: complicated software network
  - Running services: Internet based runtime

- **User participation: Crowdsourcing**
  - Wisdom of Crowd: improve services

It is about services!

Transform the Internet to service your life!
Outline

• Background
• SOA, Cloud and Crowd
• Our recent work

SOA: Service Oriented Architecture

• A software design and software architecture design pattern based on discrete pieces of software providing application functionality as services to other applications. Services are unassociated, loosely coupled units of functionality that are self-contained

Build services: reusing and composing.
Cloud computing: XaaS

Run services: elasticity, scalability, availability...

Crowdsourcing

- User participation in services
  - Requirement soliciting
  - Crowdsourced software process
  - Service rating

Improve services: building and running
The nexus of SOA, Cloud and Crowd

Future of software: Internet of Services!

SOA and Crowd

- Integration of HBS and SBS
- Crowdsourced SOA development
  - Service modeling
  - Service testing
  - Service tagging
  - Service recommendation
  - Service rating
Crowd and Cloud

SOA and Cloud

- How does cloud support the development with SOA?
  - Platform as a service for SOA
  - Service quality guarantee
- How does SOA benefit cloud services?
  - Integration of cloud services and on-premise services
  - Integration of services across cloud: interoperability
Outline

- Background
- SOA, Cloud and Crowd
- Our recent work

Our work on SOA: Web services

<table>
<thead>
<tr>
<th>Year</th>
<th>Services</th>
<th>Protocols</th>
<th>Processes</th>
</tr>
</thead>
<tbody>
<tr>
<td>2002 &amp; 2003</td>
<td>SAML Service</td>
<td>Web Service Workflow &amp; Tool</td>
<td>Web Service Transaction</td>
</tr>
<tr>
<td>2004-2006</td>
<td>XACL Service</td>
<td>Web Service Runtime 1.5</td>
<td>Web Service Reliable Message</td>
</tr>
<tr>
<td>2007-2008</td>
<td>XServices 1.0</td>
<td>XML Information Process</td>
<td>Web Console 2.0</td>
</tr>
<tr>
<td>2009-2010</td>
<td>XServices 4.0</td>
<td>WebSASE</td>
<td>Manage Console</td>
</tr>
</tbody>
</table>

863 program

HJG program

WSRuntime
BPELEngine
BPMNEngine
UDDI, SWSR
WSDT
BPELDesigner
BPMNDveloper
WSManager

WSManager

WSRuntime
BPELEngine
BPMNEngine
UDDI, SWSR
WSDT
BPELDesigner
BPMNDveloper
WSManager

WSRuntime
BPELEngine
BPMNEngine
UDDI, SWSR
WSDT
BPELDesigner
BPMNDveloper
WSManager

WSRuntime
BPELEngine
BPMNEngine
UDDI, SWSR
WSDT
BPELDesigner
BPMNDveloper
WSManager

WSRuntime
BPELEngine
BPMNEngine
UDDI, SWSR
WSDT
BPELDesigner
BPMNDveloper
WSManager

WSRuntime
BPELEngine
BPMNEngine
UDDI, SWSR
WSDT
BPELDesigner
BPMNDveloper
WSManager
Service Oriented Software Production Line

Service Oriented SPL

Developer

Business requirement → Intermediate assets → Business model → Web service → Execution model → Composite service → Composite service deployment → Software product

Business process modeling → Web service development → Composite service orchestration → Composite service test → Composite service deployment

Service Oriented Software Production Line

Service Repositories

Service Trustworthiness Tool → BP Modeling Tool → Service Encapsulation → Service Orchestration Tool → Service Running & Evolution

Modeling → Atomic services → Composite Services

Verification Tools (BPMN/BPEL) → Service Testing → Service Mtr&Mgmt → Service Simulation Tool

Service composition based development
Service4All: SOA + Cloud

- Service4All: A cloud platform for service-oriented software development
  [http://www.ow2.org/view/ActivitiesDashboard/Service4All](http://www.ow2.org/view/ActivitiesDashboard/Service4All)
  - ServiceXchange
    - A repository of service resources
  - ServiceFoundry
    - An online development environment for service-oriented software
  - Service-oriented AppEngine
    - An online runtime environment
What is NEXT?

Targeting Internet of Services

Crowd
- Big data analysis
- Service oriented Apps
- Incentive, Quality control, Aggregation
- Crowd Computing

SOA+
- Service Oriented Development
- Crowd Enhanced SOA

Cloud+
- Runtime assurance: Scalability, Availability, Consistency
- High-Assurance Cloud Computing
iCrowd@BUAA

- A mobile crowdsourcing platform

High scalability of services

A: Availability

Load Balance

C: Consistency

P: Partition Tolerance

Scale UP→Scale OUT
You can only have two of C, A and P at the same time. However, this trade-off is not yet fully understood.

Consistency and Latency

- **Inconsistency**
  - No bounded read staleness due to the eventually returned newest version
  - Write conflict usually requires semantic reconciliation
- **Latency**
  - +2s: queries/user by -1.8% and revenue/user by -4.3%
  - +500ms: -25% in searches
  - +100ms: -1% in sales
RSM-d: a quantitative model

The safety of Paxos

Write Lost
A committing write may disappear in historical records because of crash failures

Write Duplication
After inaccurate failure detection on the coordinator, the sequence number held by the write may be reused

The Simplest Case
Write Log Lost
$P_{wL} = P_{wL}$

Remove Asmp. 3

Non-uniform Write
$P_{wc} = P_{wL} \cdot P_{wNU}$

Remove Asmp. 2

Write Log Lost
$P_{wL} = P_{wL}$

Remove Asmp. 1

Write Log Lost
$P_{wL} = P_{wL} + P_{wL}$

Three Assumptions

Consistency or Latency? A Quantitative Analysis of Replication Systems Based on Replicated State Machines. DSN 2013

To summarize

Future of software: Internet of Services!
Who can build the empire of next generation software?!

Thanks!
Q&A