

Software as a Service

How about rent?

- Someone needs a software, but
 - Use few times
 - Have not much money
 - Share with the company's partner
 - ...
- Can we rent a software?
 - Decrease the TOC (total ownership cost)



SaaS

- Software as a Service (SaaS)
 - delivering software as a service over the Internet
 - eliminating the need to install
 - simplifying maintenance and support
- SaaS not only has the advantage of the ASP, but also has extra benefit
 - Fit your requirement
 - Increasing or decreasing resources on demand
 - Easy to apply, easy to use and easy to leave

So, SaaS is...

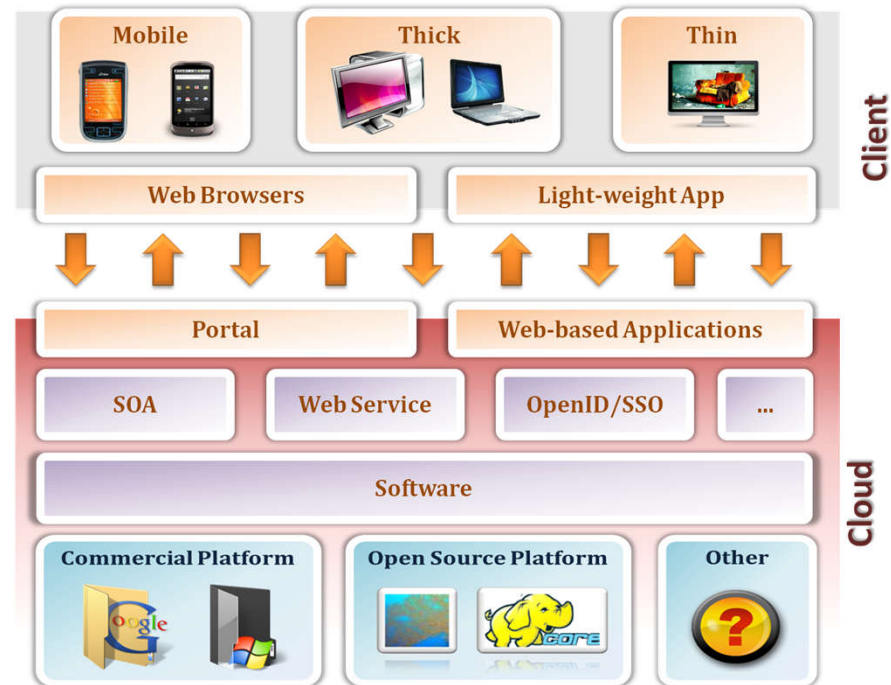
- A service model that delivers software
 - User believe that use the dedicated machine and his own operating environment.
 - Vendor can deploy many kind of software version by changing the profile.
 - Consumer meet the requirements with few extra setting.
 - Vendor can service lots of users and wide range of their needs.



Technique

Overview

- SaaS is a collection of lots of technology
 - Platform
 - Service layer
 - User interface



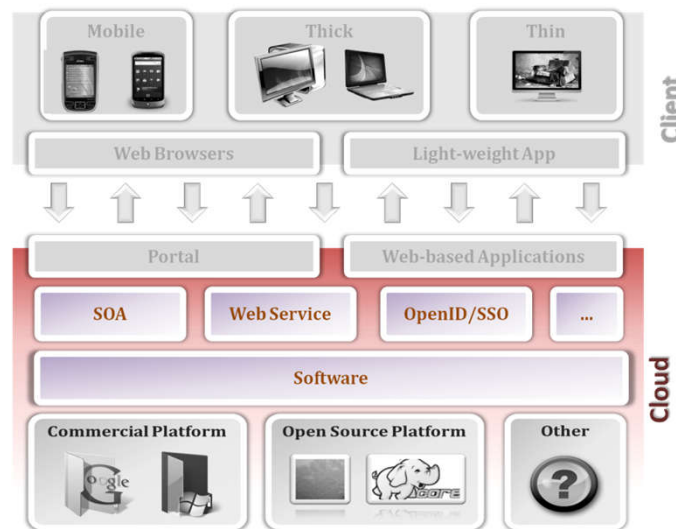
Overview (cont.)

- Platform
 - Traditional or cloud platform that provides runtime environment
- Service layer
 - Communicates or integrates between services
 - Provides SaaS properties
- User interface
 - Supplies an interesting and interactive interface
 - Reduce the difficult to use

PaaS or IaaS

Service layer

- Service layer provides many service concepts
 - How to reduce the development time and cost?
 - How to combine or integrate services and companies?
 - How to handle the access control?
 - ...etc



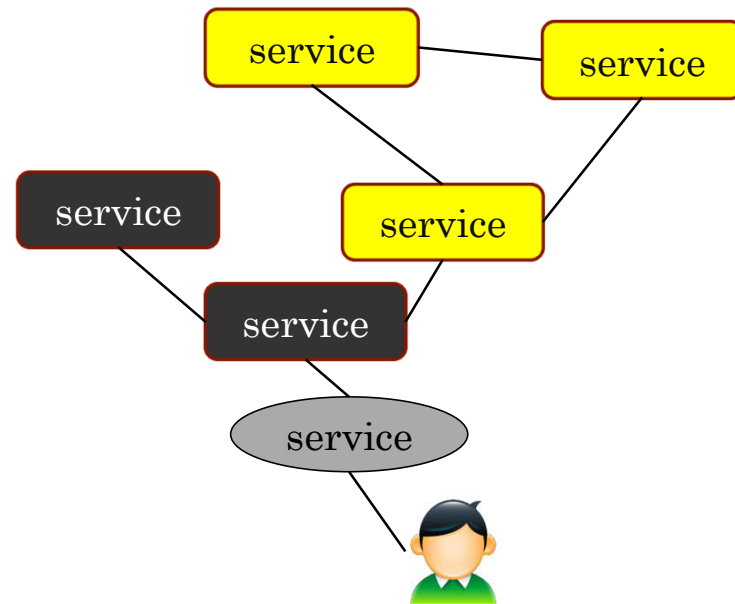
Scenario

- A company wants to design and develop a new web service
 - There are many subprojects which could collaborate with each other.
 - One or few members responses a subproject.
 - Only few months of development time



SOA

- SOA (Service-oriented architecture) is a concept that is used for reducing the cost of money, human resource and development time
 - Reuse the completed services
 - Open and standard interface
 - Service abstract
 - Loosely coupled
 - ...etc



Concept

- Reuse each service
 - Theoretically, each service can be reused every time.
 - When designing a new project, just need to implement the new features.
- Open and standard interface
 - Using the same interface (like REST, SOAP, RPC...etc) can easier to integrate, communicate and cross platform.
- Service abstract
 - User only need to know how to use.
 - Hiding the redundant information, user can use service easily with few necessary data.

Web Service

Scenario

- Service modules may not in the same machine
 - In the same rack.
 - In the same cluster.
 - ...etc.
- Communication between services is important
 - Describe the service
 - Deliver the service

Web service

- Web service is a service that responses the users' requests by
 - Using Internet technique
 - Crossing different platforms
 - Interacting, integrating and communicating with other services.
- The core concept of the web service
 - Reuse the service
 - Flexibility to integrate with other service

Web-based Service

- Web-based service is a service provided by web interface
 - May or may not interactive with other web sites.
 - May or may not provide their service.
- The benefit of web-based service
 - Cross platform – you can access the service by the browser.
 - Anywhere and anytime – in any place where you can connect to Internet.

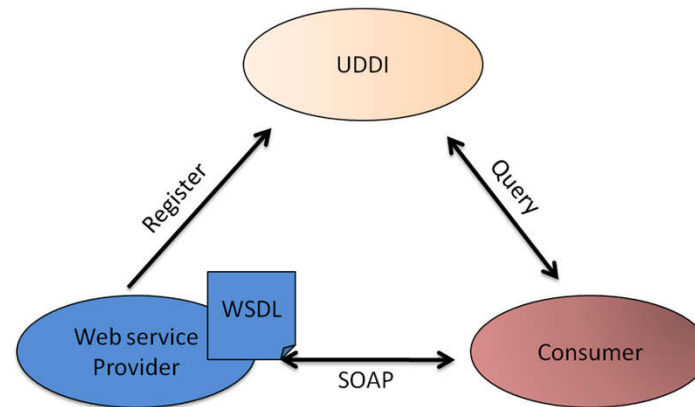
~~#Web Service~~

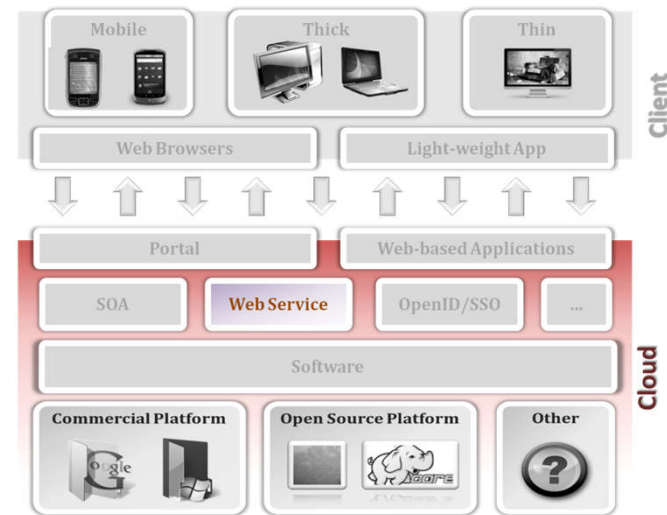
Why we need Web Service

- Time
 - The faster we provide service, the easier we get the money.
- Cost
 - Reusing the service can reduce the cost.
- Flexibility
 - Integration of services can supply more comprehensive services.

Triangle

- Web service cloud be separate into three basic parts
 - WSDL
 - The description of web service, include the service name, functionality, company's name, using method...etc.
 - SOAP
 - The method used to exchange messages between two nodes.
 - UDDI
 - A register center which can collection, integration and discovery web service.





Web service

WSDL

SOAP

UDDI

WSDL

- Web service distribution language (WSDL) is an XML file that distributes the web service, include
 - service information
 - abstractive operations
 - communication types
 - other information that can improve the effective of exchange the service.
- The host can know the web service by analysis the WSDL file.

Property

- WSDL has four advantage on exchange the web service information
 - Service abstract
 - Cross platform
 - Automatic
 - Remote

Property (cont.)

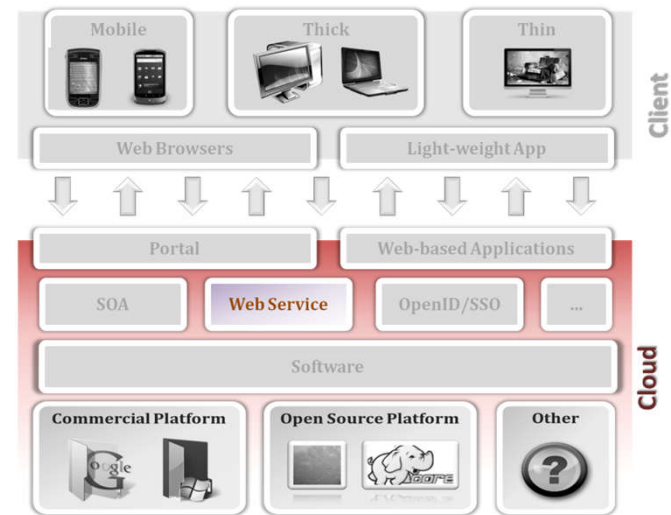
- Service abstract
 - User can access the service without knowing the service's operation logic.
 - Providers can update or patch service without notifying the user.
- Cross platform
 - WSDL uses the XML file to distribute service which can be used in any platform.
 - XML is independent on platform, OS and programming language.

Property (cont.)

- Automatic
 - There are many tools can generate or decode the XML file.
 - Reduce the cost and human resource on producing the WSDL file.
- Remote
 - By connecting to Internet and getting the correct mode of operation, user can access and call the service anywhere and anytime.

Four elements

- WSDL has four basic elements for distributing the service
 - Type
 - data structure of web service information.
 - Message
 - the parameter of exchanging message.
 - Binding
 - The network protocol using on communication.
 - PortType
 - the request/response type on communication.



Web service

WSDL

SOAP

UDDI

SOAP

- Simple Object Access Protocol (SOAP) is a communication protocol based on XML
 - By HTTP or other network protocol
 - Exchanging web services' information
 - Connecting to other web services



Core concepts

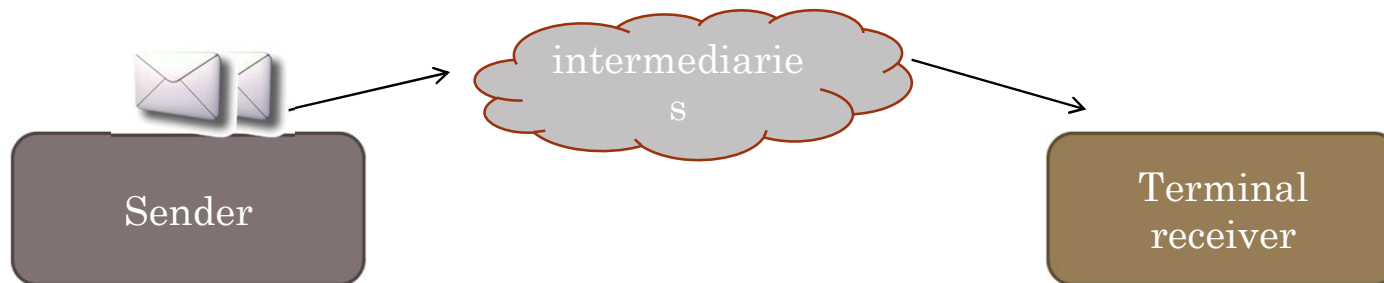
- The core concepts of SOAP are simple and extendable
 - To achieve these two concepts, SOAP omits the message framework of
 - Reliability, security, correlation, routing ... etc.
 - Instead, put these functionality into the extensibility module.
- The SOAP can be a lightweight protocol and has more flexibility.

Three basic module

- The message framework can be classify into three module
 - Processing module
 - Underlying module
 - Message construct
- And more, SOAP has a extensibility module which can add capability found in richer messaging environments.

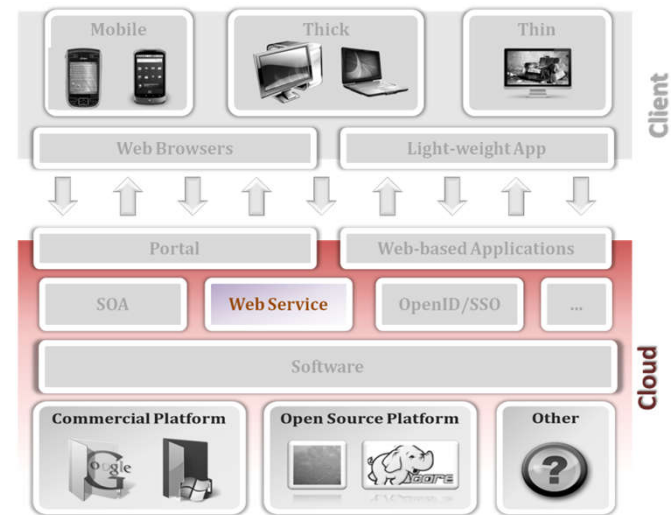
Process Module

- Processing module
 - This module defines the method of processing message
 - Message can pass through any numbers of intermediaries from sender to receiver.
 - Support many types of message exchange patterns
 - Each message is processed independent
 - This means SOAP does not recode message's state



Message Construct

- The SOAP message is specified as an XML infoset which is used in exchange message between two SOAP nodes.
- The SOAP message includes
 - Commands
 - Attributes
 - Namespace



Web service

WSDL

SOAP

UDDI

UDDI

- UDDI (Universal Description, Discovery, and Integration) is a cross-platform service which
 - Based on XML.
 - A register server.
 - Can find the service you need.
 - Can post your service.
 - Can exchange your service to another company.

Why need UDDI

- In the world
 - We cannot reach to anywhere.
 - We cannot find the service directly.
 - We cannot sell service to each person in the world.
- In the Internet
 - Network links everyone in the Internet.
 - Internet has no geographical boundaries.

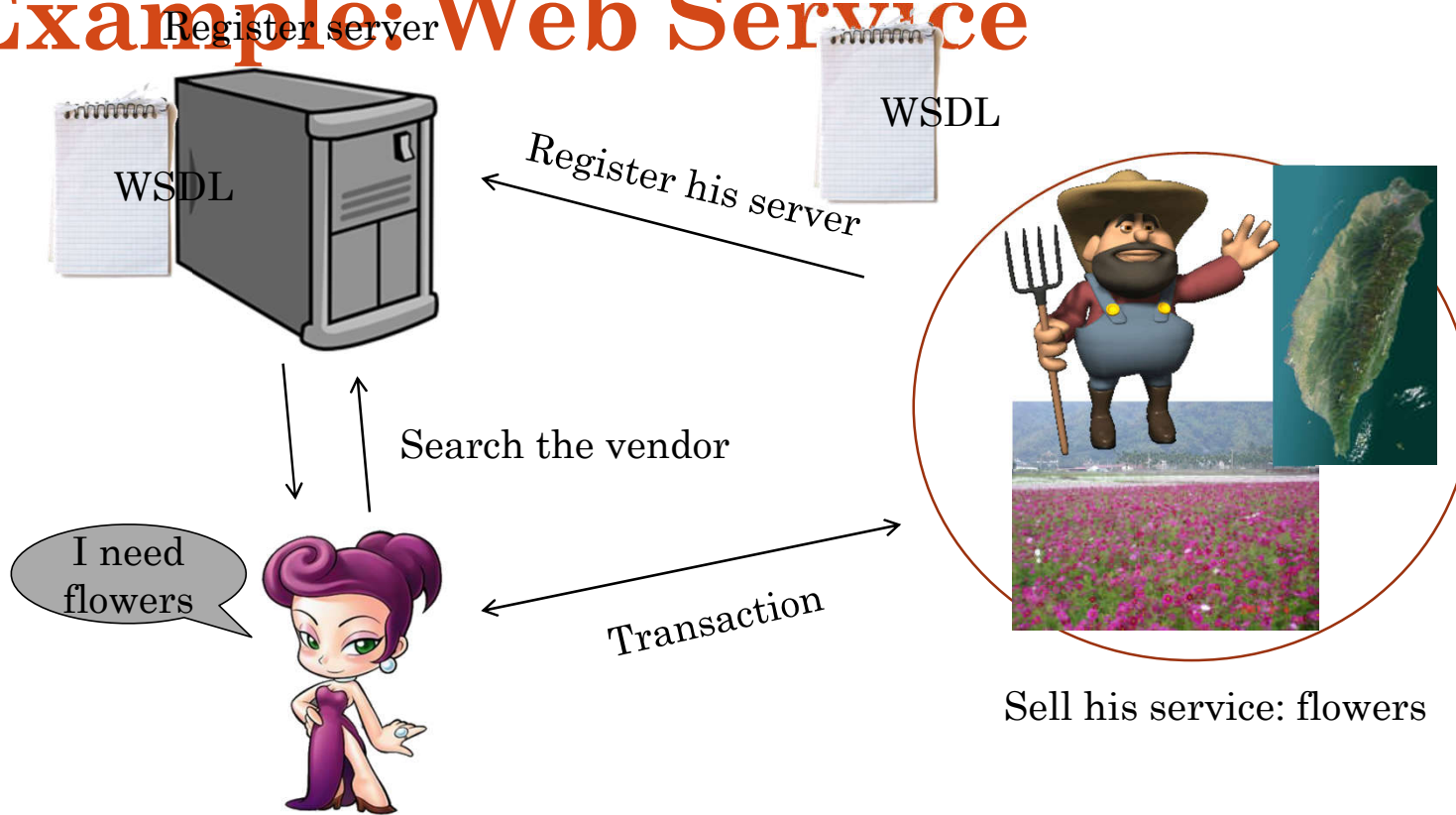
Three pages

- White page
 - Give information about the service, include name and the description of the business, which may support multiple languages.
- Yellow page
 - Provide a classification of service of business based on the standard taxonomies.
- Green page
 - Describe how to access the Web service, include the url, API and protocol.

Example: UDDI

- A company wants to post his web service
 - Find a register center
 - Design and upload their WSDL file and SOAP protocol
 - Wait until customer connect to and use their service.
- ...but he need some help
 - Search the service from UDDI register center
 - Find the service by some key-word
 - Connect to the particular web service by the WSDL file and SOAP protocol.

Example: Web Service



Service Layer

SOA

Web Service

Other service: standard

When scale up...

- Web service provides a new way to use the service and cooperate with other companies, but
 - When the users scale up, service cannot work at smooth as before.
 - When service scale up, company re-development a new service.

Solutions

- In SaaS, cloud provides a non-limitation environment that can serve users as many as possible.
- But service could be re-development many times
 - There are many services in Internet
 - Develop a new service is costly and risky
- Company should concentrate on the valuable service, not a duplication of development.

Suggestions

- There are many suggestions for this problem
 - Re-use the service
 - Easy than SOAP to connect to services
 - Reduce the complexity to use services
 - A quantitative standard of services