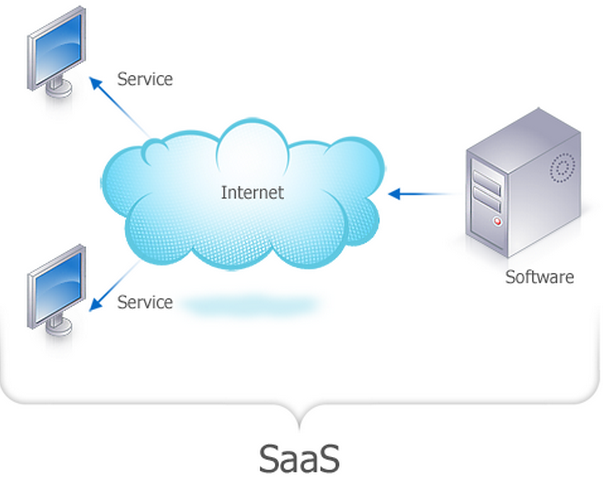
## Schedule of Course Activities: Session 32

## *[CS 519: Introduction to Cloud Computing Online-Based]*

## *[Instructor: John C. Chan]*

|  |  |
| --- | --- |
| **Overview of Session** |  |
| We will answer the following questions: | 1. Review Software as a Service. 2. API Interfaces. 3. oAuth 4. The major features. 5. … |

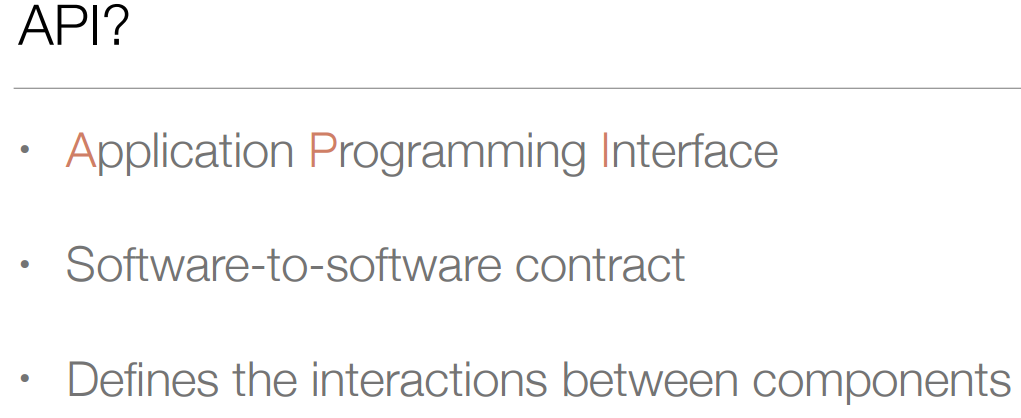


# **Software as a service**

**Software as a service** is a software licensing and delivery model in which software is licensed on a subscription basis and is centrally hosted.] It is sometimes referred to as "on-demand software". SaaS is typically accessed by users using a thin client via a web browser. SaaS has become a common delivery model for many business applications, including office and messaging software, payroll processing software, DBMS software, management software, CAD software, development software, gamification, virtualization, accounting, collaboration, customer relationship management (CRM), management information systems (MIS), enterprise resource planning (ERP), invoicing, human resource management (HRM), talent acquisition, content management (CM), antivirus software, and service desk management. SaaS has been incorporated into the strategy of all leading enterprise software companies. One of the biggest selling points for these companies is the potential to reduce IT support costs by outsourcing hardware and software maintenance and support to the SaaS provider.

The vast majority of SaaS solutions are based on a multi-tenant architecture. With this model, a single version of the application, with a single configuration (hardware, network, operating system), is used for all customers ("tenants"). To support scalability, the application is installed on multiple machines (called horizontal scaling). In some cases, a second version of the application is set up to offer a select group of customers with access to pre-release versions of the applications (e.g., a beta version) for testing purposes. This is contrasted with traditional software, where multiple physical copies of the software — each potentially of a different version, with a potentially different configuration, and often customized — are installed across various customer sites.

While an exception rather than the norm, some SaaS solutions do not use multi-tenancy, or use other mechanisms—such as virtualization—to cost-effectively manage a large number of customers in place of multi-tenancy. Whether multi-tenancy is a necessary component for software-as-a-service is a topic of controversy.



# **Application programming interface**

In computer programming, an application programming interface (API) is a set of routines, protocols, and tools for building software applications. An API expresses a software component in terms of its operations, inputs, outputs, and underlying types. An API defines functionalities that are independent of their respective implementations, which allows definitions and implementations to vary without compromising the interface. A good API makes it easier to develop a program by providing all the building blocks. A programmer then puts the blocks together.

**Web APIs**

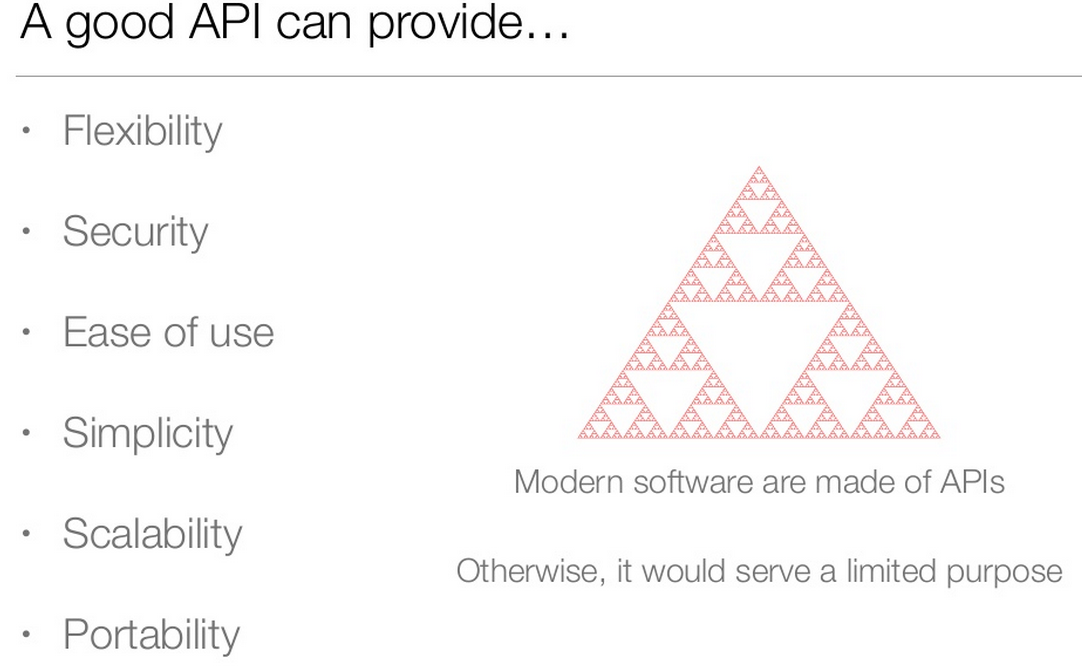
Web APIs are the defined interfaces through which interactions happen between an enterprise and applications that use its assets. An API approach is an architectural approach that revolves around providing programmable interfaces to a set of services to different applications serving different types of consumers. When used in the context of web development, an API is typically defined as a set of Hypertext Transfer Protocol (HTTP) request messages, along with a definition of the structure of response messages, which is usually in an Extensible Markup Language (XML) or JavaScript Object Notation (JSON) format.

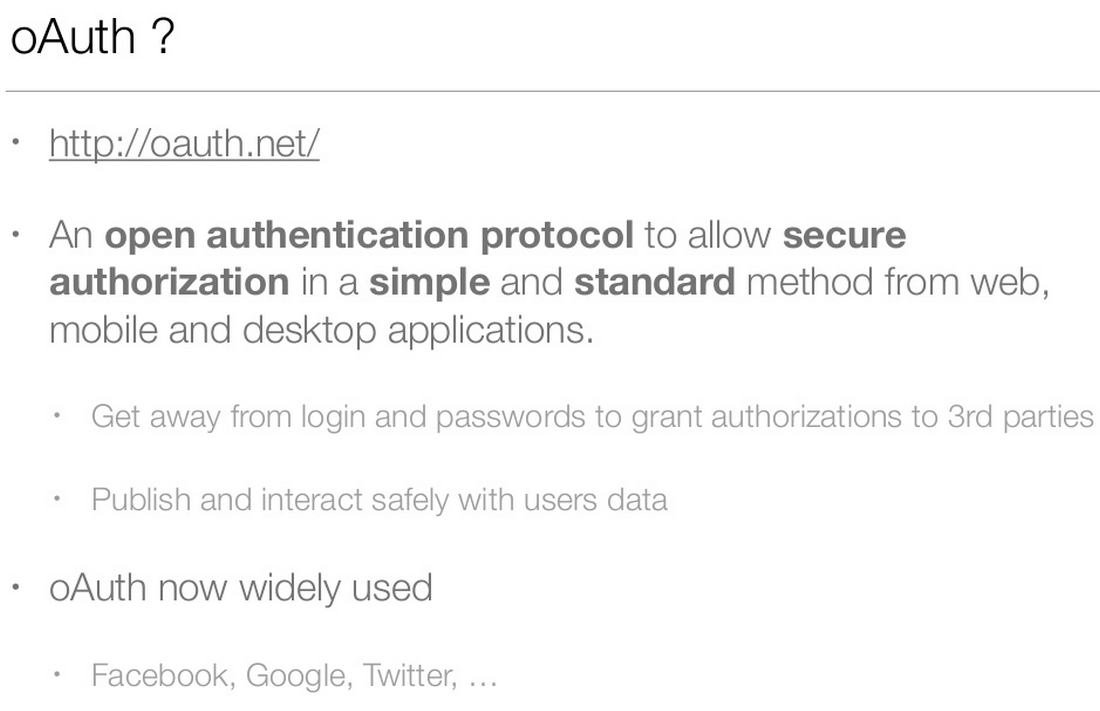
Let’s learn API from this tutorial:

<https://www.youtube.com/watch?v=7YcW25PHnAA>

NOTES:

* REST API, stands for Representational State Transfer.
* Facebook, Google Map APIs, the JSON records returned.
* API: GET, POST, PUT, PATCH, DELETE, etc.





Let’s learn more about oAuth from this tutorial vidieo:

[**https://www.youtube.com/watch?v=io\_r-0e3Qcw**](https://www.youtube.com/watch?v=io_r-0e3Qcw)

Key Take-Away:

* oAuth is an authorization framework. It does so through API interface.
* The concept of token for secure access resources under control of the application itself.
* Token is better that UserName/Password for transaction authorization.
* Token has limited scope though, e.g. Limited authorization, expiration.
* …

End-of-Class Module.

Questions? Please email to me, or post it on Blackboard.

Thank you.