Software as a Service (SaaS)

COSC349—Cloud Computing Architecture

David Eyers

Learning objectives

- Can define **Software as a Service** (SaaS)
- Describe a few popular **examples** of successful SaaS
- Compare advantages and disadvantages of SaaS to other cloud hosting models (IaaS and PaaS)
- Give a typical example of a **SaaS pricing model**
- Explain why some applications are **not suited to SaaS**
- Illustrate how SaaS may provide for **extensibility**, and how SaaS tools may integrate with other SaaS tools

Software as a Service (SaaS)

- eCommerce has been able to **operate on the web**:
  - since HTTP can submit data to servers; (HTTP/1.1 onwards)
  - since SSL (TLS) was available to secure client/server transfers
- **Local software** was typically more configurable though

- In SaaS, tenants pay to access their data, hosted on software **installed and managed by the cloud provider**
  - Often replaces the need for locally running software
  - … but various hybrid models exist, too

Example SaaS providers

- **Document management**—Google Workspace; Office 365
  - Also Overleaf (LaTeX); Zotero (citations); HackMD (our labs…); …
- **File storage** and **sharing**—Dropbox; ownCloud; Box; …
- **Web CMSs** (content management)—WordPress; Drupal
- **Communications**—Slack; Discord; MailChimp; survey tools
- **Business processes**: Salesforce.com; DocuSign; Xero; Doodle; Confluence; online VoIP systems; …
- **Software development**: GitHub; Bitbucket; GitLab
- Others areas emerging: *e.g.*, games

SaaS can support use by small business well

- Now practical for (some) small businesses to operate almost **entirely using SaaS** platforms and a laptop
  - *e.g.*, accounting, payroll, CRM, internal communications, calendaring, document management, conference calling, …
- Key cloud benefit continues to be about **variable costs**
  - Pay only for what you use, come and go as you like
- However **data management** and **lock-in** is a concern
  - Many services provide export features… but export to what?
    - Ideally export to the same software platform—FOSS facilities this

SaaS pricing, and software subscription

- Software licensing & delivery usually is **subscription-based**
  - Typically charged in **price per user per month**, *e.g.*,
    - Google Workspace—NZ$9/user/month; Slack—US$6.67/user/month
- Some organisations pushed hard for shift to subscriptions
  - *e.g.*, Adobe Creative Cloud—adds data content, not just code
    - Avoid direct competition with FOSS progress: GIMP; Inkscape; Scribus…
- Some organisations provide a range of options
  - *e.g.*, Microsoft Word is available standalone, hybrid, or web-based
- What does **"owning software" mean, in practice**?
  - Software out of support? Will bit-rot rapidly: security; OS updates…

SaaS technical requirements

- Can use SaaS **without installing software** on users' devices
- Web browsers are the typical thin client for SaaS
  - Web browsers are now ubiquitous software
  - JavaScript support can also be assumed: client-side code
    - (… this challenges some interpretations of "installing software")
- Web now well adapted to suit **smart phones**, **tablets**, *etc*.
  - Promoting **bring your own device** (BYOD) in the workplace
    - Increasing interest in partitioning users' devices into work/non-work
  - On-site software may shift to provide web access (*e.g.*, email)

SaaS storage

- Likely that **state of application** will be cloud-based
  - state of application includes documents, for example
  - however common practice to have local and cloud copies
  - even more necessary for mobile devices (that only cache files)
- Often is coupled with a **lack of a 'save'** functionality
  - Instead do auto-saving … although that requires past versions be kept, to avoid accidental corruption of records
- 'Files' iOS application is a reflection of SaaS progress
  - Provides an API to unify access to storage apps

SaaS extensibility

- Potential problem with the convenience of SaaS:
  - the software itself **becomes invisible** to the tenant…
  - so what about **customisations** that might be required?
- Much SaaS is actually highly programmable
  - May be limits on what can be used compared to local install
  - Google Drive extensions are powerful and flexible
  - Likewise apps within Dropbox, Slack, *etc*.
  - Macro programming in office-style applications
- Extensions run within SaaS, or in distributed ecosystem

SaaS integration

- Big SaaS typically tries to bring **everything into one silo**
  - *e.g.*, Office suites such as Google Workspace; Office 365; *etc.*
  - Google and Microsoft have file sync and share tools
- Tools like Dropbox must motivate their own integration
  - Dropbox considered too big to ignore within Office 365 world
  - But Google largely ignore Dropbox, since GDocs are cloud-only
- Modern SaaS will provide tools to integrate software
  - **OAuth2** standard facilitates control over delegated authorisation
  - **Storage** access for export/import/integration (*e.g.*, use S3)
  - **Logging** and **audit** of software activity

Some use cases that may challenge SaaS

- Applications with high bandwidth requirements
  - *e.g.*, editing of 8K video may **exceed WAN bandwidth**
    - … but YouTube, Echo 360, *etc*. now facilitate non-8K video editing
- **Sensitive data handling**—may need to remain onsite
  - However encrypted data handling and processing may be OK
- **Need offline access**—*e.g.*, have to work in-flight
  - Google Workspace is cloud-only, yet facilitates offline editing
- **Custom hardware**—but can still 'remote' the software
  - *e.g.*, just use a local 'plug in' to connect hardware with SaaS

Cloud-hosted gaming—new SaaS domain

- Games that use **cloud for coordination** are common
  - … but typically have lots of local software, *e.g.*, to drive GPU
- Recent announcements for **cloud-hosted gaming**
  - SaaS in that your local device just does input/output
  - output is more like playing a live video than rendering 3D
- **Google Stadia**—4K@60fps streaming of game graphics
  - Uses Chrome; Chromecast; … but questions about its future…
- **Microsoft's Xbox Cloud Gaming** (was xCloud)
  - Streamed rendering to browsers, phones, TVs, Xbox, …