Software as a Service (SaaS) ethical issues

Marco Vallini

marco.vallini@polito.it

May 2009

Contents

1 Introduction

The growing up of Internet connections, its pervasion in business companies, and the increasing of networks throughput, leads enterprise companies to offer services rather than software products. This paper analyses the Software as a Service paradigm, and highlights its related ethical issues. The section 2 describes the SaaS scenario and the differences with traditional software. The section 3 is focused on actors involved in SaaS approach. Finally, the section 4 analyses the ethical issues related to SaaS adoption.

2 Scenario

The Software as a Service (SaaS) is distinguished from traditional software paradigm primarily in that it is accessed via a web browser over the Internet, rather than installed directly into the user's computer [1]. More precisely, the SaaS is a model of software deployment, whereby a provider licenses an application to customers on demand and for use it as a service. SaaS software vendors may host the application on their own web servers, or download the application to the consumer device, disabling it after use, or after the on demand contract expires [2]. Typically an application may be composed by several services, provided by one or more Application Service Provider(s) (ASP). Examples of SaaS pioneers are SAP Business ByDesign [3], and Google Apps [4]. In particular, Google Apps provides collaboration services like shared calendar, document sharing and traditional tasks, like word-processing and spreadsheet. Thus, Google Apps would be considered as SaaS, while Microsoft Office would be traditional software. The tradition software paradigm involves business companies, enterprises, and IT professionals in debate of open source vs. proprietary software. The SaaS paradigm instead, overcomes this debate but introduces new ethical issues, analyzed later in this paper.

3 Which actors are involved in SaaS?

Software as a Service model changes relationships between software/service producers and customers, involving several actors. To understand these changes, the following paragraphs describe and correlate differences between traditional and SaaS models.

3.1 Traditional software model

The traditional model involves the software producer and the customer company. This product may be open source or proprietary and licensed to the customer. The software production process, both for open and closed source, requires IT developers and experts. Once the customer company acquires the application, purchasing or downloading it as open source software, it should install and manage the product. Typically a business application is installed on several workstations, which interact with different servers to authenticate, perform application tasks, read, share, exchange and save data. Thus the acquired application should be managed by IT staff, for doing common tasks, like patching, backup data, assure availability, privacy and security for application data and network components. For example, when a manager is out of the office, he performs the common application tasks using a VPN (Virtual Private Network).

3.2 SaaS model

The SaaS model can be divided into two major scenarios. In the first, each component of the application and data are outsourced and offered by Application Service Provider as services. As a result the customer has no direct control on application or data. In the second scenario, only a subset of application or data is outsourced, and the customer has more control on data and application. For example, a customer may outsource only public data, instead private data is stored at company. It's clear that in SaaS, the customer company requires less IT staff resources than in the traditional software model.

3.3 Actors roles in traditional software and SaaS models

In the following table are described the differences between traditional software and SaaS models for the customer company point of view. The table highlights the aspects related to human resource utilization in a customer company, between traditional and SaaS models.

3.4 Elaboration equipment and power consumption in traditional software and SaaS models

The differences between traditional and SaaS models also regard elaboration equipment and power consumption. It's clear that in traditional environments the company should purchase and maintain elaboration equipments, typically the servers which host application components and data (for database and backup). These equipment components and their power consumptions are costs. The SaaS model instead abates the number and types (no servers needed, only workstations) of elaboration equipments required for the customer company,

Activity	Actor	Traditional software	SaaS
Application and component installation tasks	IT staff	required	absent
Application and component patching tasks	IT staff	required	absent
Application component development and integration	developers and system integrators	required in some cases	absent
Securing network communications and data (VPNs, privacy etc.)	IT security staff	required	absent, or required for few tasks
Backup application data	IT security staff	required	absent, or required for few tasks

Table 1: Actors roles in traditional software and SaaS models

reducing hardware and power consumption costs. This is achieved by application service provider, using economies of scale. For example, an ASP may host several applications or different companies' data on the same server. A new qualifying technology in this field is virtualisation, that allows displacing different virtual machines (e.g. database, web server, etc.) on the same physical host, providing isolation between different environments.

4 Ethical issues

The adoption of SaaS model, against traditional software, introduces some ethical issues related to several aspects. Firstly the outsourcing approach leads customer companies to buy a set of services licensed by application service provider. As a result, the application elaboration equipments, like servers, became not necessary. As described before, this condition allows reducing the hardware costs, because the company elaboration requirements may be satisfied using common workstations. The related ethical issues, described in the following paragraphs, regard:

- security and privacy

- service accessibility and availability

- ecology, power consumptions and waste management

- job opportunities

4.1 Security and privacy

The adoption of SaaS also introduces security and privacy ethical issues. This paper analyses the following categories:

- privacy and confidentiality of stored data

- privacy and confidentiality of data flow

- data availability and backup

- ASP company profile

4.1.1 Privacy and confidentiality of stored data

SaaS solutions store the company's data, (e.g. documents, contacts, notes, billing information, etc.) on remote servers, rather than on the company's computers. Given that one of the most important objectives is to protect the confidentiality of the information contained in these files. The ASP manages data of different business companies, and should provide isolation, to assure that each company only access to own data. Typically these concept are expressed into Service Level Agreement (SLA), but is not simple verifies these requirements. For example, if the SLA defines that any stored data should be ciphered, and data is stored on ASP database, how does the business company can verify this? In addition we can suppose that the SAP perform data ciphering but, in this case, who can manage cryptography keys? Or who can decrypt company's data? We observe that there are different data categories, with different relevance. For example, a company can store and manage business data (e.g. credit cards data, orders, payments, etc.) that is relevant, but not life critical. However other companies (e.g. pharmaceutical industries) or institutions (e.g. local health units) may manage health care data that, is very relevant and life critical. An interesting ethical problem would be the analyses of outsourcing data related to health, or other of critical nature.

4.1.2 Privacy and confidentiality of data flow

The company's data should be exchanged and processed by different applications and services. Thus we should consider that each interaction that involves data (a data flow) between SaaS components should respect privacy and confidentiality properties. The business company can simply verify these properties evaluating interactions between its workstation and SaaS services, for example checking if the data flow is protected using SSL/TLS or IPSec protocols. But how the company can verify if the data flow regarding SaaS components is protected? This condition is not simple to verify. As a result, the company can rely on SLA and should trust the ASP, without having a proof.

4.1.3 Data availability and backup

Data availability, backup policies and formats are crucial aspects, and could introduce ethical issues. When a company outsources its data the related issues are:

- which data is available?

- which backups are available?

- what are backup format and its compatibility?

Let's imagine that the company outsources its data to an ASP. The ASP analyses which set of data are more accessed and decide to provide on-line availability only for a subset of data (more accessed data). If the SLA admits this condition, this could be legal. Another relevant issue regards backup availability. In particular, the SLA could describe that backup is provided every day, but the company can directly access to backups? If the company can access to its backups, what is backup format? This can be proprietary or not compatible with common software, or company's systems. Finally, if the company decide to cancel SaaS subscription, will the company get back its data?

4.1.4 ASP company profile

The previous considerations are focused on data security and availability; however the choice of outsourcing services should takes in account the ASP company profile. So how a business company can rely on ASP? The answers should consider several aspects. Firstly we can consider aspects related to financial conditions, evaluating company investments and funding origins. Secondly we can evaluate company strategies in terms of local and global job opportunities and development. The ethical issues, related to job opportunities, are discussed later in section 4.4.

4.2 Service accessibility and availability

The [1] article discusses and highlights the accessibility and availability differences between traditional and SaaS models. The access to traditional software is often limited by both hardware compatibility and licensing restrictions. For example, traditional business software may only be compatible with a PC running a particular operating system (e.g. Windows XP).

SaaS solutions are web-based and, as a result, they're typically compatible across multiple platforms (e.g. Mac, PC, Linux, etc.) and browsers (e.g. Internet Explorer, Firefox, Safari, etc.). In addition, a SaaS solution may also be compatible with mobile devices (e.g. iPhone, Smartphones in general), whereas traditional software cannot be used on mobile devices, or can only be used by purchasing a separate mobile version of the software. SaaS also has the advantage of not being limited to a single installation, as is often the case with traditional software. Thus, with SaaS a user could use the web-based interface from home, office or any other location with Internet access, without having to purchase additional licenses for each computer. This can also be useful for business continuity purposes for example, in case of disaster, an user can be back working as soon as a web-enabled computer is available. The introduction of web-based approach and the full application of W3C guidelines [6] are optimal solutions to provide accessibility and universal access to people with disabilities. The [6] are a set of best practices for all Web content developers (page authors and site designers), and for developers of authoring tools. The final objectives are promote accessibility, and make web content more available to all users, whatever user agent they are using (e.g., desktop browser, voice browser, mobile phone, automobile-based personal computer, etc.), or constraints they may be operating under (e.g., noisy surroundings, under- or over-illuminated rooms, in a hands-free environment, etc.). As a result, the web-base approach and the W3C guidelines application are the way to solve ethical issues related to accessibility for people with disabilities.

4.3   Ecology, power consumptions and waste management

The SaaS model also leads to ethical considerations about ecology and recycling aspects. The introduction of virtualisation and cloud computing can be considered the enabling technologies to achieve green economy objectives. More precisely, ethical considerations regard the following categories:

- hardware equipment

- power consumption

- waste management

4.3.1   Hardware equipment

The virtualisation technology allows displacing several virtual machines with different operating systems and services on the same physical host. Nowadays if a company wants to deploy two different services (e.g. web and email) it provides two virtual machines on the same server, instead of using two different physical hosts. However, different companies have different systems and physical hosts, for displacing own services. The SaaS approach extends virtualisation concept, providing virtual services aggregation; a SAP can displace on the same server different companies' services (e.g. using a virtual machine for each service). As a result, the necessary hardware equipment to provide services is significantly minor. More practically, the [5] interview suggests that using virtualisation techniques, for a typical project, reduces the numbers of servers by a factor between 10 to 1, and 20 to 1. That turns

into, on average, a 40 percent overall cost savings, and an 80 percent reduction in hardware purchases.

4.3.2   Power consumption

The reduction of hardware equipment and the services aggregation allow achieving more energy efficiency. Less equipment also reduce the power consumption related to data center servers and cooling system. Consequently this efficiency leads to reduce the CO_2 emission, decreasing air pollution, and preserve planet conditions.

4.3.3   Waste management

The reduction of servers, required to provide services, consequently allows buying less hardware and throwing away fewer machines, when they reach their end of life. As a result, this directly impacts on the waste management, and related costs: collection, transport, processing, recycling, or disposal.

4.4   Job opportunities

Finally, the lack of application software installed at the company remove most of the issues related to components installation, integration, developing, patching activities, backup activities, data privacy and security. These activities, always performed by human resources (IT staff, developers, IT security staff, etc.) are no longer required and, as a result the job opportunities may be decrease significantly. Another relevant issue is related to job market globalization. Often a SAP enterprise employs IT staff coming from global job market. Instead, in the traditional application software model, the small or medium companies attempt to employ staff coming from local market. As a result this strategy enhances the development of local economies, in terms of job opportunities and for the local allied industries.

5   Conclusions

A subset of the proposed ethical issues can be mitigated using technologies and business strategies. This section has attempted to investigate which are the available solutions. Some of security and privacy ethical problems can be solved outsourcing only non-sensitive data (strategy), or ciphering data using an in-company key management system (technology). For example, a company that sells products through e-commerce service can only outsource the electronic catalogue, but store orders using in-company system. This approach can also interesting for job opportunities: if a company stores orders internally, it should employ an IT staff to manage the system. The issues related to data flow security, that regards interaction between SAP components, are not easy to solve. Typically a company should specify its own security requirements into SLA contract. However, for small or medium companies this is not simple, because the SAP has more contractual power than customers. A technology-based solution is monitoring interactions between components to verify their security properties. However the SAP could not allows company to monitor its components; in that case, the problem can't be solved. Data availability and backup, as for data flow security, should be specified into SLA contract. Once the SLA contains availability and backup specification, checking if backups are available and which is the format, is a quite simple task.

References

[1] *FYI: Software as a Service (SaaS) for Lawyers*, `http://www.abanet.org/tech/ltrc/fyidocs/saas.html`

[2] *Software as a Service*, Wikipedia, `http://en.wikipedia.org/wiki/Software_as_a_service`

[3] *SAP Business ByDesign*, `https://www.sme.sap.com/irj/sme/en_GB/home/`

[4] *Google Apps*, `http://www.google.com/a/help/intl/it/index.html`

[5] *Virtualization - The Foedus Interview*, `http://www.treehugger.com/files/2007/03/virtualization_interview.php`

[6] *Web Content Accessibility Guidelines 1.0*, `http://www.w3.org/TR/WAI-WEBCONTENT/`