Business Continuity Solutions for SaaS Customers

Conference Paper *in* Lecture Notes in Business Information Processing · June 2011

DOI: 10.1007/978-3-642-21544-5_3 · Source: DBLP

2 authors:

Tommy van de Zande
Utrecht University
3 PUBLICATIONS **71** CITATIONS
SEE PROFILE

Slinger Jansen
Utrecht University
174 PUBLICATIONS **2,073** CITATIONS
SEE PROFILE

Business Continuity Solutions for SaaS Customers

Tommy van de Zande and Slinger Jansen

Dept. of Information and Computing Sciences,
Utrecht University
{T.Jacobusmeergenaamdvande Zande,R.L.Jansen}@uu.nl

Abstract. Organizations are increasingly adopting SaaS-solutions in favor of traditional on-premise solutions, because of the advantages in terms of cost reduction, implementation time and scalability. Business continuity of these SaaS-solutions is often neglected, even when business processes that depend on the SaaS-solution are critical. This paper addresses business continuity for SaaS-solutions by identifying and evaluating different business continuity solutions to protect customers from the risk of their SaaS-provider going bankrupt. Two solutions; 'SaaS-escrow' and the 'SaaS-guarantee-fund', are evaluated in expert interviews and a survey. The conclusions of this research are that there is a need for SaaS business continuity solutions, SaaS continuity solutions are not frequently employed, and that the two solutions presented here are favored equally by a panel of business managers.

Key words: Business Continuity, SaaS, Escrow

1 Introduction

Software as a service (SaaS) is a form of software deployment, that delivers software on a subscription basis through the internet. With SaaS, companies no longer have to buy or develop a complete software solution up front, instead they *rent* it. Pricing models can vary, but generally customers pay on a subscription-basis or on a usage volume-basis [1]. SaaS is rapidly growing in popularity. Recent developments in Internet technology and broadband adoption enabled online software to serve as a true desktop software replacement. Also the recent economic and financial crises have played their part in showing the advantages of SaaS, because of its low upfront investment and high scalability. Because of these advantages in comparison to traditional software licensing models, a large number of small businesses and start-ups are already using SaaS solutions for some time now, and even large institutions and corporations are migrating their data from on-premises software to externally developed and hosted software.

Service Level Agreements are legal documents where customer and provider agree on the quality of service, by quantifying minimum quality of service [2]. For SaaS, these mostly discuss availability and response time. There are some problems that are generally neglected, like what happens when a SaaS-provider

goes out of business. If a company is using mission-critical software that is hosted off-premises, it could get into big problems when the provider decides to pull the plug. The business continuity risks include (in order of importance):

R1: Lose access to data
R2: Lose access to the application itself
R3: Lose support and maintenance

These problems are considered deal breakers by decision makers and counter measures need to be taken to assure continuity, before SaaS can truly serve as a replacement for offline software.

For shrink-wrapped software, business continuity can be arranged using a source-code escrow [3]. With source-code escrow, the developer stores its source code with a trusted third party, the escrow agent. If this developer goes bankrupt or otherwise defaults on his support and maintenance obligations, the customer requests the escrow agent to release the source-code. If the escrow-agent confirms that a valid *release condition* has been met, he will transfer the source-code to the customer.

In the late 90s of the previous century, application service providers started hosting applications off-site, offering some of the same advantages that current SaaS-providers offer. In the end those traditional ASPs were not able to deliver reliability and quality standards demanded by business customers [4]. The business models of most traditional Application Service Providers (ASPs) were fundamentally different from current SaaS business models; traditional ASPs repackaged existing legacy software and offered it off-site, so customers still had to buy a license to use the software [5, 1]. After the burst of the *dot-com bubble* in the year 2000, many of those traditional ASPs faced financial problems, and some even bankruptcy, leaving their clients with no access to their software [6, 7]. The new generation of ASPs, who offer true SaaS-solutions, differ in a way that they do not resell usage rights for existing enterprise applications. They develop their own web-based applications on a new multi-tenancy design paradigm, which makes serving multiple clients more scalable and cost-effective [5].

If SaaS is going to serve as a more cost effective and flexible replacement for shrink-wrapped software, it is imperative that there is a clear way for both customers and providers to arrange a viable continuity solution. Several scholars identify the need for research attention to the business aspects of SaaS, and mention the risk of SaaS-providers going bankrupt as a serious issue [8, 9].

The goal of this paper is to give a *state-of-the-art* report on available continuity solutions for SaaS. We discuss the necessity of such continuity solutions and compare different initiatives. Since there is no clear solution to the aforementioned problem, the research is explorative. First the requirements needed for a successful SaaS business continuity solution are identified. The identified requirements are based on the results of five semi-structured interviews: four interviews with CEOs from several Dutch SaaS vendors and one interview with the CEO of a Dutch Escrow agent. During these interviews, we distinguished two different SaaS continuity solution. These solutions are described and compared with each other. We also compare these existing solutions with the requirements

that have been identified. Then we discuss the necessity of SaaS continuity arrangements. And finally, a small survey is carried out among SaaS providers and customers.

The paper is built up as follows; Section 2 describes the interviews and presents some initial findings. Section 3 discusses what risks should be covered to provide SaaS continuity, and presents the requirements for a SaaS continuity solution. In section 4 we discuss what solutions are currently being offered. Section 5 discusses the necessity of business continuity solutions for SaaS. Section 6 presents the results of the survey. In section 7 we conclude the paper.

2 Interviews

Because SaaS Business Continuity is a new and unknown topic, we started our research by conducting five semi-structured interviews with several people in the SaaS and continuity business. These interviews provided data for both the requirements for a continuity solution as well as details of the two existing continuity solutions.

2.1 Interview Design

The five interviews were conducted with four CEOs of several (small) Dutch SaaS-providers and the CEO of one Dutch escrow-agent. The interview participants were selected pragmatically. Two were approached within the network of the authors. Two others were found at a conference on SaaS and cloud computing. They were all approached because their SaaS-services targeted other businesses. The interviewed escrow-agent was referred by one of the SaaS-providers. The interviewee selection process threatens the validity of the research, in that the participants were selected based on availability and willingness to participate. However, due to the method used to find participants we dare say with some certainty that the interviewees' responses were relevant, on-topic, and addressed the topic with sufficient experience (avg. 5 years) and insight. The SaaS-provider interviews discussed themes such as customer demand for business continuity solutions, use of business continuity solutions, what business continuity solution they would prefer and what would happen to their customers if the provider suddenly disappeared, what problems they would face and what data they would lose. We also asked the interviewee's if there were any other solutions to provide business continuity. The interview with the escrow-agent was conducted to gather detailed information about how their SaaS-escrow solution works. All the interviews took approximately one hour and data was recorded by taking notes. The notes were later transcribed and sent to the interviewees for verification. Three of the SaaS-provider interviews and the escrow-agent interview were conducted on locations at the companies. One SaaS-provider interview was conducted by phone. All interviews took place during March-June 2010. Table 2.1 shows some characteristics of the different companies where we conducted the interviews.

Table 1. Overview of the interviewed companies along with the type of product they offer and the approx. number of customers.

Company	Product	Customers	Business Coninuity Solution
Provider 1	SaaS CRM	100	None
Provider 2	SaaS Planning suite	39	Escrow for specifc customer
Provider 3	Several solutions	60	None
Provider 4	SaaS ERP software	80	Escrow (optional)
Escrow agent	Saas-Escrow	20	N.A.

2.2 Initial Results

Two of the interviewed SaaS-providers (Provider 1 and 3) explained that they have thought about business continuity solutions, but in the end did not go through with it, because almost none of their customers demanded such a solution. They stated that a lot of customers do not worry about the financial stability of their SaaS-provider, but that it was also possible that they didn't understand the risks. Provider 2 did use an escrow-solution, but only with one of their customers, because they were the only one demanding such a solution. Provider 4 offered escrow as an extra service, but none of their current customers actually applied for it.

According to the SaaS-providers, there is not much demand for SaaS business continuity from the customers. Some of the SaaS-providers believed that customers do not fear the continuity risks because of the following fact; a SaaS-provider, like any other subscription-based service, has clear vision and control over its finances. A SaaS-provider can predict with a high amount of certainty what his revenue will be over a certain period of time. When the SaaS-provider uses one-year contracts, then he knows how much revenue he will make for the next 12 months at any given moment. So the only way that he could get into trouble is if his costs will go up unexpectedly, which is rare because most of his costs are also on a subscription-basis. Added to those clear costs is the trust that, when bankruptcy does occur, a Bankruptcy Trustee will keep the software running for as long as possible because it is a revenue producing part of the company. Section 5 elaborates on this assumption.

The clear and stable financial situation of SaaS-providers might explain the absence of business continuity worries among SaaS-customers. However, demand for business continuity solutions does exist. The escrow-agent stated that he currently sees an increase of demand for their SaaS coninuity solution. The explanation of SaaS-providers that their customers do not have to fear bankruptcy might be true for most customers, but some simply can not bear the risk.

3 Guaranteeing Business Continuity for SaaS

In this section, we explain the business continuity risks and requirements for a successful business continuity solution. These risks and requirements are based

on the interview data. The risks and list of requirements were created during the first interview. These risks and requirements did not change throughout the further interviews, although they were reformulated and reshaped during the second and third interview. The fourth interview corroborated the results from the previous interviews.

The SaaS-model is fundamentally different from the traditional software-licensing model. The difference is that the customer does not possess the object-code on-premises but, instead, accesses the application on a remote server, using the internet. This remote server is managed by the SaaS-provider, either on-site or by using a hosting-provider. The actual hardware where the software and data reside is out of the customer's reach and control. Some SaaS-solutions even include content from third-party content providers in their SaaS-software. A customer does not have anything to do with all those external parties, and commonly they do not even know that external parties are being employed. The customer pays its SaaS-provider for access to the software, the SaaS-provider in his turn pays the different parties involved to deliver its service. Together with the interviewed experts, we identified several requirements for a business continuity solution.

The main goal with a business continuity arrangement is the assurance that the customer continues to have access to his SaaS application and data, even if the SaaS-provider disappears. To make the continuation of access and data work several key elements should be covered. The most important element is the customer's data. Even if access to the application has been suspended, with a recent backup a customer at least does not have to worry about losing his data, and he can start migrating towards an alternative solution, and only lose access to his application for a couple of days or weeks, depending on the size of the data and type of application. Losing access to the application would still be a major problem for any organization, but without access to (a recent backup of) the data, problems would be much worse; imagine a company losing all its data that resided in their CRM system. The company would not be able to service their current customers or process new leads. Losing access to CRM data could be disastrous for a lot of companies. So the first step towards business continuity would be the ability to acquire regular backups of the data.

The next step towards a more complete business continuity agreement would be an agreement with the hosting-provider, in such a way that they ensure they will continue hosting the application even when the SaaS-provider gets into financial difficulties. Such an agreement could be arranged by a SaaS-customer itself, but that would not work if there are more customers hosted on the same server, which is generally the case with SaaS. Therefore, a logical step would be to arrange this hosting continuity agreement with a separate legal entity. This legal entity can either be a commercial escrow-agent, or a foundation/fund founded by the customer(s) or SaaS-provider themselves. Such a separate entity can also provide some additional services next to simply continuing hosting (and providing the funds to do so). They could offer support for the application when the SaaS-provider fails to do so. This hosting continuity is a kind of insurance,

and will be cheaper if arranged with multiple SaaS-providers at the same time, because the chance that all of the SaaS-providers fail at the same time is lower than the chance that one of them fails. Some SaaS-providers also use third party content or services in their applications. Sometimes this content is free, but frequently the SaaS-provider pays the content provider for the content. For better continuity these third party providers also have to be included in the business continuity arrangement. The last step for complete business continuity is continuing support and maintenance for the application, to help customers with possible problems and keep the application running.

To summarize, the requirements for a complete SaaS business continuity solution are (in order of importance):

1. **Own Backup**: Every SaaS customers should be able to download all of its data.
2. **Hosting Insurance**: A third party should create an arrangement with the hosting provider to continue hosting even if the SaaS provider fails.
3. **Arrangement with content providers**: If the SaaS application contains (paid) content from third parties, they should also continue providing the content.
4. **Support and maintenance for the application**: If the SaaS provider disappears, the customer also loses support. A third party could try to continue support for the application.

A solution that meets these requirements, should be able to effectively protect customers of a SaaS-provider when it goes bankrupt or otherwise out of business. Two solutions that were identified during the interviews are presented in the next section.

4 Available Solutions

Even though SaaS business continuity guarantees are not common, several companies, like the escrow-agent we interviewed, are already offering solutions. In this section these solutions are discussed, and compared with the requirements we identified for a successful business continuity guarantee. Data about the SaaS-escrow solution came from the interviewed escrow-agent, completed with data from websites of several escrow-agents. Data concerning the SaaS Guarantee fund came from the interviewed CEOs of Provider 1 and 4. The SaaS-providers also pointed out the downsides for both solutions.

4.1 SaaS-escrow

A common solution for SaaS business continuity, SaaS-escrow, is offered by existing escrow-agents, who already offered source-code escrow for traditional software. As the interviewed escrow-agent pointed out, most escrow-agents have added a 'SaaS-escrow' service to their product portfolio. SaaS-escrow usually is

a modified version of the regular source-code escrow of the escrow-agent. The modification generally consists of the addition of a data back-up with the deposit of source-code. More complete escrow solutions also provide 'continuation of hosting', where they arrange an agreement with the hosting provider, that whenever the SaaS-provider gets into problems, the escrow-agent takes over the financial obligation towards the hosting-provider. The hosting provider in return promises that they will continue hosting the SaaS-application and data under any circumstance. Escrow-agents differentiate their solution by offering different extra services for SaaS-escrow, like delivering support and maintenance of the escrowed application when the escrow is released. The escrow-agent takes over support and maintenance for a predetermined amount of time. During this time, customers have the ability to migrate their data to another more permanent solution. SaaS-escrow solutions can be arranged on two different levels. The first one is a three-party arrangement with the SaaS-provider, the SaaS-customer and the escrow agent. In this arrangement the individual customer is the only customer who is able to access the application when the escrow is released. But when multiple customers demand an escrow-arrangement, the second arrangement makes more sense; a two party 'master-contract' arrangement between the SaaS-provider and the escrow-agent. In this arrangement there is no limit on how many customers become a beneficiary of the escrow-arrangement, to benefit from the arrangement only depends on each individual customer if they want to sign up for it (and pay the price of course). Such a master-contract arrangement is initially more expensive than a single three-party arrangement, but as an advantage it is much easier to add new customers to the arrangement, and spread the costs over all the participating customers.

With SaaS-escrow, as opposed to traditional escrow services, the initial purpose of storing the source-code and releasing it to the customer on certain release-events is less important than the continuation of application access. Most SaaS-customers would not have any use for the source-code, because they probably do not have the hardware and infrastructure to deploy the software application on-premises. As an added value, the escrow company can offer support and maintenance for the SaaS application, by storing documentation and remaining in contact with key-persons involved with the software-maintenance at the SaaS-provider. So with SaaS-escrow, the escrow-agency acts more like an insurance company for hosting costs than a storage facility for sensitive information. This also creates a possible risk for the business continuity of the escrow-company itself. If the SaaS-provider grows in size, the cost for hosting the application grows accordingly. That way, it could become too expensive for the escrow-company to take over hosting-costs if a big SaaS-provider goes bankrupt.

Another possible problem with SaaS-escrow arrangements is that the solution is general and standardized, so for some specific SaaS-solutions the escrow-solution simply would not work or only cover a part of the business continuity problems. For example, typical escrow solutions do not offer support for SaaS-applications, which use a lot of third-party content in their application, because they only cover continuation of payment towards the hosting provider, but not

the payment towards third-party content providers. Another example is a SaaS-provider who uses a lot of different hosting-providers to host their application, for example to provide better reliability and speed for customers around the world. The escrow-company then should sign a contract with every single one of those hosting-providers to be able to continue hosting the application for every customer.

4.2 SaaS Guarantee Fund

The CEOs of Provider 1 and Provider 4 both explained another possible solution to guarantee business continuity: a SaaS Guarantee fund. The SaaS Guarantee Fund is based on the idea of the so-called *Travel Guarantee Fund*, which exists in several countries. Such a Travel Guarantee Fund covers the risk for travelers who book a trip with a travel-agency or tour-operator which goes bankrupt before or during the actual trip. The Fund provides customers of a participating travel-agency with (financial) protection against such risks, so that customers are guaranteed that their trip is paid-for even though the travel-agency defaults. An adaption of such a fund could function as a business-continuity guarantee solution for SaaS-providers. SaaS-providers have a clear image of their financial situation over the coming months. They know what their costs will be to pay every third-party involved in running the SaaS application for upcoming months. With this financial forecast in mind, they could set up a fund with a budget large enough to cover those costs for several months. The fund then arranges an agreement with all those third-parties, to continue their services towards the SaaS-provider under any circumstance. Since the fund is a different legal entity than the provider itself, it is not affected by financial problems or bankruptcy of the provider. In case of bankruptcy or severe financial problems, the fund can take over the payment towards all third-parties for a few months, during which customers have time to migrate their data towards another solution, or during which the SaaS-provider can make a new start again *"Storing the code and data at a third party could be dangerous regarding theft of IP and setting up a guarantee-fund is not that hard to achieve and has the advantage of (expected) lower costs."* is what one of the survey respondents answered when asked which arrangement he thinks works best.

The guarantee fund could initially be small and only support one single SaaS-provider, so it can be perfectly tailored to support that single solution (and its customers of course) on all (business continuity) aspects. To lower costs and efforts, multiple SaaS-providers could set-up a fund together, lowering the required cash-deposit per provider because it is unlikely that the fund has to cover for all the participating providers at the same time. But a fund for multiple providers also has its disadvantages, for example: the fund will be less customizable towards every specific SaaS-solution. Another disadvantage is that when one of the participating SaaS-providers fail, then the others feel that they pay for its failure.

Currently, there are no known SaaS-guarantee funds with multiple participators. The problem probably lies in the initial start-up of such a fund. Who

takes the initiative and invests the initial time and money in it? SaaS-providers are commercial companies, and they justify their investments and projects with a business case which predicts a profitable outcome. A SaaS-guarantee-fund for multiple providers does not have any apparent extra benefits for the SaaS-provider who initiates the fund. Some companies have started a SaaS-guarantee-fund for their own solution though. An apparent party to set up and manage a multiple SaaS fund would be a (software) trade association. They can provide the fund along with possible certification for participating SaaS-providers.

4.3 Comparison

An advantage of the SaaS-escrow solution is that it is easy to set up, because the solution is an existing package for which providers only have to sign up once. Most escrow-companies already exist for several years, and have the required legal and technical knowledge to provide a reliable business-continuity solution. Another advantage are its transparent costs. Because escrow solutions are relatively standard, the costs are known in advance. A downside is that SaaS-escrow is a standard solution with less customizability than a custom SaaS-guarantee-fund, a SaaS-provider who uses a lot of different hosting parties and content providers would have a hard time finding a suitable escrow-solution. Also, escrow-solutions are more expensive than SaaS-guarantee-funds because of overhead costs and the for-profit nature of escrow-companies. Survey respondents who preferred the escrow-arrangement used arguments as: *"The SaaS Escrow guarantees that the source code and data are stored and will be given to the customer when things go wrong, whereas a Guarantee Fund will only give help to customers (missing the actual 'guarantee' as given by the other arrangement)"*, *"They [The escrow company] have the legal expertise available"* and *"Because it is easier to set up and can be arranged on forehand"*.

An advantage of a SaaS-fund is that it can be set-up for a single SaaS-solution, so the provider remains completely in control of who has access to its source-code and other intellectual property, while still providing a workable business-continuity guarantee. Also, because of its non-profit nature and low overhead costs, all of its income will be used to serve its core activity; provide continuity, instead of overhead costs like marketing and management. Arguments favoring the SaaS-fund were: *"Without the capital knowledge of the technical staff, the code/data is of very little use."*, *"Storing the code and data at a third party could be dangerous regarding theft of IP and setting up a guarantee-fund is not that hard to achieve and has the advantage of (expected) lower costs."* and *"It is the only option that can cover the complete infrastructure, including all third parties and resellers."* Table 2 summarizes the possible advantages and disadvantages.

When we compare both solutions to the previously mentioned requirements on how a complete business continuity solution should work, we conclude that both solutions can cover the basics: they both offer data backups and continuation of hosting. SaaS-escrow does not offer extra continuity agreements for

	SaaS-escrow	Guarantee Fund
Advantages	Easy to arrange Legal knowledge Clear costs Experience	Complete control Customizable for specific solution (Expected) lower costs
Disadvantages	Expensive External party	Requires more effort from provider Responsibility stays with the provider No prior experience

Table 2. A table showing the advantages and disadvantages of the different solutions.

other third-parties like content providers or extra hosting-partners. The SaaS-fund could contain all of those options, but in the end the features of a SaaS-fund depend on how the fund is implemented.

5 Necessity of SaaS continuity arrangements

There are several ways to ensure business continuity with SaaS. The only question that remains is: is it necessary? This question has to be answered by each (potential) SaaS-customer individually. Several authors doubt the necessity of the traditional source-code escrow [10, 11, 12] for shrink-wrapped software, because few escrows are actually released. So it is only logical to doubt the necessity of SaaS business continuity solutions as well. Of course, the model of SaaS versus that of shrink-wrapped software is completely different, and so are the risks at stake. With SaaS, if things go wrong, they could have disastrous consequences for some customers, because they could lose access to their software as well as their data. But the chances of a SaaS-provider going bankrupt and leaving its customers without any access to their application or data from one day to another is in fact small, when for example considering the interests of a bankruptcy trustee. *"A great deal depends on the application itself – how critical is the data in the application? What are the workarounds I can use to back up my information without incurring a lot more work?"* one respondent answered on how he thinks about business continuity arrangements.

5.1 The Bankruptcy Trustee

The interview results show that many SaaS-customers have not given much thought to business continuity risks, and frequently rely on a SaaS solutions without proper business continuity guarantees. Others, as pointed out by the CEO of Provider 4, simply did not worry that they lose access to their application because they believe in a simple yet effective assumption, that is based on the SaaS business model itself; the SaaS model consists of a constant revenue stream. When a SaaS-provider files for bankruptcy, a Bankruptcy Trustee will always keep that constant stream of revenue flowing because it can be used to pay off

creditors. To keep the revenue flowing, he will have to keep the SaaS application running. The Bankruptcy Trustee would cut off departments like marketing and R&D, but will keep the core services running.

The danger is that this trust in the bankruptcy trustee is based on an assumption, albeit a logical one. A Bankruptcy Trustee is not obliged to keep the service running, and could decide to liquidate all assets instead, leaving its customers without their software. Also, a SaaS-provider can stop its services even though it did not go bankrupt. For example, the Californian-based Platform-as-a-Service provider Coghead, which provided an online hosted platform to create enterprise database applications, announced in February of 2009 that their intellectual property assets were acquired by SAP, and that they would stop supporting the platform within the next month [13]. Customers had one month to develop a new application on another hosted platform and migrate their data towards it. Some companies see these risks as an ultimate downside for outsourcing their IT to an external service provider [14], and use it as an argument to stick to the traditional on-premises software.

6 Survey

To gain some insights in customer preference, and in an attempt to verify the identified requirements, we carried out a small survey. The survey focussed on gaining insight in the number of companies who utilize business continuity solutions and tries to identify a preference for one of the two different continuity solutions.

6.1 Survey Design

In an online survey, we asked several IT decision-makers and SaaS providers about their thoughts on business continuity with SaaS. We asked them if their SaaS-provider provided a solution for business continuity, how important they think such a continuity solution is, and which of the previously presented continuity alternatives they think is best. The survey results are based on 20 respondents. We selected participants by posting a message in several SaaS-user and provider groups on LinkedIn. We also asked the members of the Dutch CIO platform to participate. The survey first asked respondents wether they were a customer or provider of a SaaS provider. Then the respondents were asked if their SaaS-solution already offered a business continuity solution. Respondents who indicated that they are a customers of a SaaS-solution were asked wether they would consider a SaaS-solution if it did not provide a business continuity solution. Then the survey asked respondents to rate several aspects of SaaS-products/providers on a scale of importance ranging from *"not important"* (depicted as 0 in the graph) to *"extremely important"* (depicted as 3.0 in the graph). The different aspects were: size of the company, location of the company, price of the SaaS-solution, financial situation of the SaaS provider, technical continuity solutions, business continuity solutions and data export abilities. Finally, we

presented a short description of the two business continuity solutions discussed in this paper; the SaaS escrow solution and the SaaS guarantee fund, and we asked respondents to indicate which one they preferred and explain why.

6.2 Survey Results

From the 20 respondents, 50% claimed to provide a SaaS-solution, the other 50% claimed to be (professional) SaaS-users. Only three of the in total 20 respondents (or their provider) currently offered a complete business continuity solution like SaaS-escrow or a guarantee fund. The other 17 respondents either simply answered *"no"* or stated that they were still looking into it. Five respondents stated that their provider offered data-export abilities as a continuity solution. These results are quite surprising when we look at the results of the second survey question, which we asked to the group that stated they were (potential) SaaS-users: *"Would you consider a SaaS-solution if it does not provide a clear solution for business continuity?"* Seven out of ten respondents answered that they *"could not take such a risk"*, while the remaining three respondents answered that it depended on the specific company and SaaS-solution.

The combined results of the ranking of importance of several SaaS aspects are visualized in Fig. 1. As expected, the location of the company is perceived less important than aspects like price, financial situation and technical continuity, since the access to SaaS-applications is location independent. Respondents rated data export abilities as the most important aspect when selecting a SaaS-solution. This is in line with our findings in the previous chapter, but it is surprising to see that even though it is perceived as extremely important, not every SaaS-solution offers this possibility. The next most important aspect is technical continuity. That is not surprising, because the likelihood of technical problems is much higher than that of business continuity problems. The respondents rated business continuity solutions as the third most important aspect of a SaaS-solution, with almost the same score as price and financial stability. This is quite surprising because business continuity solutions are not common in SaaS-solutions, based on the survey results which showed that only 3 out of 20 SaaS-solutions offer some kind of business continuity solution. So there is a big difference between how important people think business continuity solutions are and the number of SaaS-providers actually using a business continuity solution. There was no significant difference between the ratings of SaaS-providers and customers.

Exactly 50% of the respondents preferred the escrow-solution, while the other 50% preferred the guarantee-fund alternative. Some arguments in favor of the escrow-solution were: *"Because it's easier to set-up and the arrangement is active immediately"* and *"they have the legal expertise available"*. Arguments in favor of a SaaS guarantee fund were: *"Without the capital knowledge of the technical staff, the code/data is of very little use"* and *"Storing the code and data at a third party could be dangerous regarding theft of IP and setting up a guarantee-fund isn't that hard to achieve and has the advantage of (expected) lower costs"*. There was nog significant difference between the solution-preference

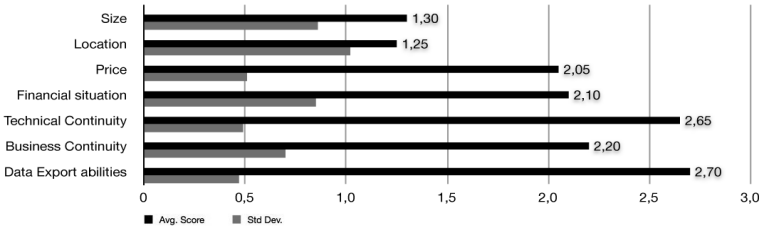

Fig. 1. A graph showing the average score of several aspects related to SaaS-solutions on a scale of importance (from 0 to 3.0), along with the standard deviation.

among SaaS-providers and SaaS-users. Several respondents explained that they prefer an escrow arrangement for standard SaaS-solutions, but that they would prefer a custom guarantee fund for more complicated solutions with a lot of third parties, because a guarantee fund has the ability to *"cover the whole chain"*.

To summarize, the survey showed that people think business continuity solutions are important for SaaS-providers, but currently not many SaaS-providers actually use a business continuity solution. The survey showed that both of the discussed solutions are seen as viable options for business continuity guarantees, with equal votes, but each with different advantages and disadvantages.

7 Conclusions

A company going bankrupt will always be a risk for a customer, no matter what kind of business it is in. A business continuity solution could work to make the consequences for a customer less disastrous, but it is hard to provide complete protection in every scenario. In most cases, customers of SaaS-providers can find comfort in the fact that because of the SaaS business model, keeping the service running has the highest priority even if the company goes bankrupt, because it provides a continuous revenue stream. In any case, the customer at least has to make sure that he has access to his data and be able to acquire a backup. This is the most important step towards business continuity. A customer should ask himself how problems with the SaaS provider would affect his own business. If a customer would get into serious problems with its own business continuity if the SaaS provider fails, then a business continuity arrangement makes sense.

As our survey showed, most providers and customers think business continuity arrangements are important, but not many providers are currently offering a business continuity solution. This probably has to do with the fact that SaaS is a relatively new phenomenon and that there is no 'best practice' yet. With this paper we hope to give some clarification on this topic, and help clarify the different options available to arrange business continuity. Several issues that could influence the validity of the survey results is that the number of respondents is quite small and it did not distinguish between different SaaS-solution. It is possible that several respondents were using the same solution. However, the

goal of this paper is not to identify how many companies currently use business continuity solutions, or point out which solution is preferred. Instead, this paper calls for attention on the topic and provides insights in the risks at stake and the two possible solutions.

The types of business continuity solutions we discussed both have their pros and cons, and there is no one *best* solution. Because the SaaS model, in its current form, is relatively new, there are no practical examples to assess the effectiveness of both solutions in real life. There are no known cases where one of the two continuity arrangements were ever actually put into effect yet.

Theoretically both the escrow-solution and the fund-solution should function as a reliable solution. The big difference between the two is that the escrow-solution is a commercial solution, which could be more expensive because the escrow-company needs to make a profit, but offers a complete and ready to use solution with professional (legal) support. The fund-solution can be cheaper to set-up, but is more time consuming and requires a lot of effort from the SaaS-provider itself. What is the best solution depends on the type of SaaS solution and personal preferences of both the provider and its customers. We think that when business continuity solutions are needed, for most standard SaaS solutions the escrow version is preferred because of its simplicity and low effort requirements. When the SaaS solution is more exotic or needs more specific arrangements with many third parties or for a difficult infrastructure, the fund-solution appears to be a better alternative.

Acknowledgements. The authors would like to thank the interviewed CEOs and survey respondents, for their valuable insights. We would also like to thank the anonymous reviewers, and especially Pasi Tyrväinen, for their valuable comments and suggested improvements.

References

1. Abdat, N., Spruit, M., Bos, M.: Software as a service and the pricing strategy for vendors. In Strader, T., ed.: Digital Product Management, Technology and Practice: Interdisciplinary Perspectives. Advances in E-Business Research (AEBR) Book Series. IGI Global (2010) 154–192
2. Hiles, A.: Service Level Agreements: Panacea or Pain? The TQM Magazine **6**(2) (1994) 14–16
3. Freeman, E.: Source Code Escrow. Information Systems Security **13**(1) (2004) 8–11
4. Dubey, A., Wagle, D.: Delivering software as a service. The McKinsey Quarterly (2007)
5. Kaplan, J.: SaaS: friend or foe? Business Communications Review **37**(6) (2007)
6. Currie, W., Seltsikas, P.: Exploring the supply-side of IT outsourcing: evaluating the emerging role of application service providers. European Journal of Information Systems **10**(3) (2001) 123–134
7. Chen, M., Chen, A., Shao, B.: The implications and impacts of web services to electronic commerce research and practices. J. Electron. Commerce Res. **4**(4) (2003) 128–139

8. Marston, S., Li, Z., Bandyopadhyay, S., Zhang, J., Ghalsasi, A.: Cloud computing - The business perspective. Decision Support Systems **51**(1) (2011) 176–189
9. Armbrust, M., Stoica, I., Zaharia, M., Fox, A., Griffith, R., Joseph, A.D., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A.: A view of cloud computing. Communications of the ACM **53**(4) (2010) 50
10. Mezrich, J.: Source Code Escrow: An Exercise in Futility. Marquette Intellectual Property Law Review **5** (2001)
11. Denson, W.: Source Code Escrow: A Worthwhile or Worthless Investment. Rutgers Bankruptcy Law Journal **1** (2002)
12. Helms, S., Cheng, A.: Source code escrow: Are you just following the herd? `http://www.cio.com/article/187450/Source_Code_Escrow_Are_You_Just_Following_the_Herd_` (2008)
13. Savvas, A.: Coghead customers left high and dry despite sap acquisition. `http://www.computerweekly.com/Articles/2009/02/20/234935/Coghead-customers-left-high-and-dry-despite-SAP-acquisition.htm` (2009)
14. Spiotto, A., Spiotto, J.: Ultimate Downside of Outsourcing: Bankruptcy of the Service Provider, The. American Bankruptcy Institute Law Review **11** (2003)