Applying Variability Management in the Development of a Complex SaaS System: Real Experience and Findings

Leticia González Folgueira, Oscar Pedreira, Ángeles Saavedra Places and Fernando Silva-Coira
Facultade de Informática, Universidade da Coruña, Campus de Elviña s/n, A Coruña, Spain

Keywords: Variability, SaaS, Home Care System.

Abstract: In this work, we present the complex problem of the variability that has a SaaS application for Home Care System that we have developed. We detail how was the management of this variability that was increasing as the number of clients grew (currently more than 130 companies use our application of Home Care Service in different municipalities). Thanks to the fact that the different needs of these companies and municipalities were managed from the beginning as points of variability, we were able to undertake the development of variations in the functionalities of the developed software. We also present the variability management tool that we have developed to give autonomy to the platform's marketing team. Finally, we highlight the lessons learned and present an approach to evaluating the cost of managing variability.

1 INTRODUCTION AND MOTIVATION

Delivering software as a service (SaaS) has become a major trend in the last years. This approach allows the customers to forget about aspects such as complex software deployments, maintenance, and managing and maintaining a hardware and network infrastructure for their information systems. This approach has also many advantages for the developer companies, such as a greater control on the software and more flexible licensing schemas. However, it also presents some challenges.

Typically most customers share a large set of features, but they must be able also to customize the software in some way, for example, adding or removing functionalities, or parametrizing the software so its behavior changes to adapt to the particular customer needs. This usually results in a complex parametrization of the software that must be handled carefully. A promising approach for dealing with variability analysis and modeling in these environments is that of feature-oriented analysis (Apel, 2013; Siegmund, 2017; Galster, 2016). This approach has already been considered in previous work, such as (Moens, 2012; Lizhen, 2010; Mietznet, 2009; Weiping, 2009).

In this paper, we present a real experience in using variability modeling and management to analyze and model the variability of a complex web system delivered as a service, more specifically, a system for managing home care services for dependent people. In addition to present the system and the result of the variability modeling, we also present results on the real use of the system and the different variability points we have identified and developed.

This project has been developed in Spain, where all city councils have the obligation to present a home care service to dependent people. These people are evaluated according to their medical and economic situation by the municipal technicians, who finally assign them a number of weekly hours of attention and a price to pay for them (which can range from 0€ to a growing percentage of the real cost according to their economic situation).

However, city councils, in general, do not provide this service directly, but they subcontract it to specialized companies. These companies are the ones that really provide the service and the city council gives indications and monitors the work of that service. This subcontracting is done through public bidding and the subcontracted companies compete in the hourly price that they will charge for the service.

The management of this service is extremely complex for both companies and city councils. On the one hand, the hourly price of the service the municipality pays to each company depends on the conditions of the contract and can also specify a price for different days (weekdays, weekends, holidays,

Folgueira, L., Pedreira, O., Places, Á. and Silva-Coira, F.
Applying Variability Management in the Development of a Complex SaaS System: Real Experience and Findings.
DOI: 10.5220/0007234103910397
In *Proceedings of the 14th International Conference on Web Information Systems and Technologies (WEBIST 2018)*, pages 391-397
ISBN: 978-989-758-324-7

nights, etc). On the other, companies have to pay full salaries to their assistants (not for the hours they worked, which makes it critical for them to plan the service that minimizes the displacement of assistants between homes). In addition, each user pays the hours of service actually received each month with a percentage of their cost. The cost varies from user to user according to their economic situation.

In addition to charging and billing, the service organization itself is complex because there are dependents who should always receive the service (they depend on it to eat, get out of bed or to have a minimum hygiene). On the other hand the auxiliaries have the right to vacations, days off and sick leave, however, replacing them is complex as it sometimes requires increasing the number of work hours of other colleagues, which is only possible within certain legal limits. Besides, dependents have periods of hospitalization or travel to their family home which means interrupting the service (and stop paying) but the company must continue to pay the salary to their assistants and always minimize periods without work or of displacement between users since they do not charge for them.

Finally, municipalities want to be informed of everything at any time, especially the hours of arrival and departure of each assistant to each domicile and when a dependent changes his assistant, as frequent changes mean a low quality of service.

In order to solve this complex problem, we developed an application that is being marketed as SaaS and is currently in use in more than 130 Spanish city councils, especially in Galicia. This application offers a company profile and a city council profile to access the data of each municipality, so that the application itself serves as a means of communication between them. The check-in in each domicile is achieved by calls from the number of the domicile to a special phone number of a digital switchboard and is reflected in the database available to the city council and company. On the other hand, a mechanism is offered so that the company can efficiently plan the organization of the service and the scheduling of its assistants. The city council can also access this planning and see which assistant provides service to each user each day.

Although the mechanics of work are similar in all companies and municipalities, the reality is that points of variability began to emerge from the beginning. This is due to the fact that different companies and city councils have different protocols of action and, logically, they wanted the management application to support their way of working.

Aware that the variability management is in general a preferable option to the development of independent versions that are installed independently to each client, we assumed from the beginning that each variant in a point of variability should be implemented in the code as an alternative to the flow of execution, dependent on a control parameter of that point of variability. These parameters are inserted in the database associated to each client. Thanks to making this design decision from the beginning, we have been able to undertake the growing development of new functionalities that have been bringing new points of variability as described in Section 3.

The rest of the paper is organized as follows. Section 2 describes some aspects of the Home Help Service management necessary to understand the points of variability. Section 3 describes the different points of variability integrated in our application, both for the city council and the company. Section 4 presents an empirical analysis of the real deployment of the system and how the different variability points are being used. Finally, Section 5 presents the conclusions of the paper.

2 PROBLEMATIC OF THE VARIABILITY OF THE HOME CARE SERVICE

A fundamental concept in the management of the home care service is that of a "bag of hours", that is, the number of hours that each assistant works in a given period. Managing these data for each assistant allows the company of a home care service to automatically have up-to-date and detailed information about the hours hired and worked by their employees, in addition to other data of interest. Calculations of the bag of hours are held weekly at the end of each week. The bag of hours is shown as a table where its main columns are:

- Nº of hired hours: Number of hours assigned to the employee for his current contract.
- Nº of worked hours: Number of hours actually worked by the employee. For this, the hours of the services carried out in a period of time are added. Certain factors are also taken into account, such as absences or services canceled at the request of a user.
- Nº of complementary hours: Number of overtime hours worked beyond the time hired and paid separately.
- Nº of permission hours: Number of paid hours (permission, holidays, public

holidays, sick leave, etc) and unpaid permissions.
- Balance of hours: Difference between hours contracted and hours actually worked and paid permits.

In the generation of these data, in addition to the information of the services worked by each employee, a multitude of parameters are involved. Further, each city council and company calculate the bag of hours differently, taking into account certain factors of interest. Therefore, the bag of hours is a challenge due to its high complexity and variability.

Another complex part of the system is the billing of services. The company and the city council generate invoices for the services in a certain period of time. Therefore, it is necessary to generate both the invoice that the company charges the city council for providing the service and the invoice that users must pay to the city council, although in some cases companies are allowed to directly bill users.

3 POINTS OF VARIABILITY

The points of variability of this system follow a specific hierarchy. The parameters that allow the different users to adapt the behavior of the software to their needs can be grouped in parameters for city council (those that affect the city council and the companies), parameters for the company (those that affect the company and, if it is the case, each auxiliary of the company) and parameters for each auxiliary assistant of a company (each auxiliary can have a different value for these parameters).

Secondly, there are several kind of variability points according to the possible values that can be selected. They can be classified into three types: option Yes/No, option of a list of values and configurable value.

3.1 Parameters for a City Council

The following parameters can be configured by each city council:

VP_1. Cohabitation Unit: it allows to include if other members of the cohabitation unit are also beneficiaries of the Home Care Service. Possible values are Yes/No.

VP_2. Visualize the Planning of the Assistants of Other Companies Providing the Same City Council. In certain city councils, they contract several companies for the home care service. In order to improve the coordination between them, this parameter allows a company to visualize the planification of the assistants of the other companies. Possible values are Yes/No.

VP_3. Report Template: Template to be used in all communications generated automatically between the city council and the company and vice versa. In this case, the application offers a generic template for all communication or one template for each type. Communications are an essential part of the home care service and it is necessary to notify certain events that occur during the provision of the service by the company.

VP_4. Document Language: Languages of documents generated in PDF format. Currently, the supported languages are Spanish, Galician and Catalan. Being a public service, municipalities require you to use any of their official languages. For example, all councils in the region of Galicia must be able to use Galician and Spanish.

VP_5. Companies Invoice Users: Users must pay part of the service received. Such services are billed by the city council or directly by the company. It indicates whether companies will bill the users or it is a responsibility of the city council. The possible values are Yes/No.

VP_6. Type of Billing to Users: It indicates how the hours to bill each user are counted. There are different values that can be selected:

- By carried out service.
- By maximum of assigned hours.
- By assigned hours.
- By assigned hours (excess borrowed).

VP_7. Computation for the Billing of Public Holidays to Users: It indicates that public holidays are billed (if service is available that day). Usually, services on holidays are charged as special services and have a different price than one business day. Options are: “Only public holidays”, “Only Sundays, Saturdays and Sundays”, “ Public holidays and Sundays” and “Saturday afternoons, Sundays and public holidays” (in the last case is necessary to indicate the start time of the holiday on Saturday afternoon).

VP_8. Rounding of Total Hours Invoiced to Users: In monthly invoicing, indicate whether fractions of hours are invoiced or rounded to a integer number. Possible values are Yes/No.

3.2 Parameters for a Company

The following parameters can be configured by each company:

VP$_{9}$. Telephonic Check-in: It allows assistants to check-in and check-out their service by making a call to a virtual phone number. With this information the city council and the company can know the absences of their assistants, or if an assistant has arrived late (or has left before) an domicile, for example. It is also used by the application to generate invoices. Possible values are Yes/No.

VP$_{10}$. Calculation for Displacement Time: Assistants attend several users in the same day, so they have to move from one domicile to another during their workday and several times. During the planning process this time is taken into account when assigning services to assistants and it is also shown in the bag of hours. This parameter allows to indicate the means of transport that is used for the displacement. This option has two values; driving and walking.

VP$_{11}$. Computation for Public Holidays for Bag of Hours: It indicates that public holidays are computed for generation of the hours bag. Options are “Only public holidays”, “Only Sundays”, “Saturdays and Sundays”, “Public holidays and Sundays” or “Saturday afternoons, Sundays and public holidays” (in the last option is necessary to indicate the start time of the public holiday on Saturday afternoon).

VP$_{12}$. Time to Bill in Each Service: The list of options are:

- Always schedule time: The planned duration is invoiced for each service.
- Carried out time (only if it is lower than schedule time): The actual duration is invoiced for each service, in case the duration is less than the planned duration, for example, a service of 60 minutes is finally billed 55 minutes. Otherwise the planned duration is invoiced, for example if the 60-minute service was finally completed in 65 minutes, the 60 planned minutes will be billed.
- Always carried out time: The actual duration is always billed, regardless of whether it is lower or greater than the planned time.

VP$_{13}$. Margin Minutes: In the billing types "Carried out time (only if it is lower than schedule time)" and "Always carried out time" a margin is established from which the time carried out is counted and not the planned time. For example, if for a 60-minute scheduled service, 59 minutes are finally performed, it will probably be interesting to invoice the 60's as well. However, if 50 minutes are done, it is already interesting to invoice 50. This margin could be any positive integer.

VP$_{14}$. Field to Include in the Schedule Report for Each Assistant: Assistants receive a report from the company with their current planning. It shows the work schedules and the users that must be attended. Report can included the following fields for each user who attends, which actually have been modelled as four different boolean points of variability:

- Name and surname (**VP$_{14\cdot1}$**)
- Telephone number (**VP$_{14\cdot2}$**)
- Register number (**VP$_{14\cdot3}$**)
- Total number of kilometers between services (**VP$_{14\cdot4}$**)

VP$_{15}$. Type of Billing to City Council: It indicates how the hours to bill each city council are counted. There are different values that can be selected:

- By carried out service.
- By maximum of assigned hours.
- By assigned hours.
- By assigned hours (excess borrowed).

VP$_{16}$. Computation for the Billing of Public Holidays to City Council: It indicates that public holidays are billed (if service is available that day). Options are: “Only public holidays”, “only Sundays, Saturdays and Sundays”, “ Public holidays and Sundays” and “Saturday afternoons, Sundays and public holidays” (in the last case is necessary to indicate the start time of the holiday on Saturday afternoon).

VP$_{17}$. Rounding of Total Hours Invoiced to City Council: In monthly invoicing, it indicates whether fractions of hours are invoiced or rounded to a integer number. Possible values are Yes/No.

VP$_{18}$. Time Control in the Scheduler: It indicates whether prevents confirm plans that do not meet exactly the scheduled time of the intervention plan. Possible values are Yes/No.

3.3 Parameters for an Assistant

VP$_{19}$. Calculation of Hours Worked on Public Holidays: It indicates if the hours worked on holidays will compute double for that assistant. Possible values are Yes/No.

VP$_{20}$. Calculation of Hours in the Last Week of the Month: It indicates how to reflect in the bag of weekly hours the division of a week in two months (end of a month and beginning of the next). There are two possible values:

- Real: Hours worked, permits and others account for the part of the week in which they actually take place.

- Proportional: A proportional distribution is made in the two parts of the week in which the calculation is divided.

VP_{21}. Calculation of Rest Time after 6 Continuous Hours of Work: It indicates whether a rest time will be counted in the bag of hours as time worked. Possible values are Yes/No.

VP_{22}. Minutes of Rest Time to Compute: In the case of computing a rest period in the bag of hours, this parameter indicates the number of minutes to be computed. The value is a positive integer value.

4 EMPIRICAL ANALYSIS

In this section we present an analysis of the deployment of the software for different customers over a period of two years. In this analysis we will focus especially in the use of the variability points we have described in previous sections for the different user types (councils, companies, and workers).

As we have already explained in the introduction of the paper, the system has been developed as a service, so variability analysis and modelling has not been applied as a way of obtaining (or automatically generating) different versions of this software, but as a tool for identifying and modelling all the parameters that define the variability points required by each user type. From its deployment, it has been commercialized to a total of 100 councils and 132 assistance companies, which has resulted in 3554 workers using the platform to reflect the result of their daily work (see Table 1).

Table 1: Number of users of each user type in the real deployment over a period of two years.

User type	# of users
City council	100
Assistance company	132
Auxiliar	3554

Although these data may seem strange at first, remember that the service provided by this system can be purchased by city councils, assistance companies or a combination of both. That is, an assistance company may use the system to manage the daily operations even if the city councils for which it works are not using the platform (or vice versa).

In the analysis of the real deployment we have especially focused on how the different variability points we have developed have been used. Table 2 shows for each variability point and for each of its possible options, the number of users who are currently using each option. The table structures in three sections the variability points for the three user types supported in the software: councils (variation points from 1 to 8), companies (variation points from 9 to 18), and auxiliary assistance workers (variation points from 19 to 22).

The description of the variability points provided in the previous section states the different values for each of them. Most of them can take just two values (yes or no), while others can take more values (for example, VP_{22} can take five different values).

Interesting results can be seen in the data shown in Table 2. In most variation points one of the values is used by a large portion of the customers, so that value acts as a kind of default value. More interestingly, here is an important difference between, for example, the variation points VP_1 and VP_2. Both can be used by city councils but, while 19 users use the first option for VP_2, only one uses that value for VP_1. This triggered an important question for a real software development company. According to these data, we concluded that introducing the variation point VP_2 in the system was worthwhile, since a 81% of the customers are using the default option for this point, while a 19% are using the other option. However, in the case of VP_1 only one customer is using an option different from the default value. Was it worthy/profitable to introduce this variation point in the system? It is not so clear in this case. From the data, it just seems that this variation point does not respond to a clear need of many options for an average customer, but for something that may have been identified as a variation point but that was really a customized development for the specific need of one customer.

Without the pretension of establishing a formal and complete metric to evaluate the usefulness of introducing a variation point in a system, we have further analyzed these data by computing for each variation point the quotient between the smallest and the largest values between the number of users who chose each option for the binary variation points. Figures from 1 to 3 graphically represent the results we obtained for each user type. As we can see in the results, some variation points can be considered more valuable than others from the point of view of the number of users using each possible option for them. Although this would require a greater effort, we think that this may be a basis for establishing a way of deciding if a given development must be included in

Table 2. Number of users taking each value (Vi) for each variation point (VP_j).

Val. / VP	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17	V18	V19	V20	V21	V22	V23	V24	V25
VP_1	1	99	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
VP_2	19	81	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
VP_3	-	-	99	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
VP_4	-	-	-	-	79	21	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
VP_5	19	81	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
VP_6	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	100	0	9	0
VP_7	-	-	-	-	-	-	-	-	98	24	1	9	-	-	-	-	-	-	-	-	-	-	-	-	-
VP_8	1	99	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
VP_9	30	102	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
VP_{10}	-	-	-	-	-	-	103	29	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
VP_{11}	-	-	-	-	-	-	-	-	84	39	-	9	-	-	-	-	-	-	-	-	-	-	-	-	-
VP_{12}	-	-	-	-	-	-	-	-	-	-	-	-	132	-	-	-	-	-	-	-	-	-	-	-	-
VP_{13}	-	-	-	-	-	-	-	-	-	-	-	-	-	132	-	-	-	-	-	-	-	-	-	-	-
$VP_{14\text{-}1}$	55	77	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
$VP_{14\text{-}2}$	46	86	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
$VP_{14\text{-}3}$	1	131	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
$VP_{14\text{-}4}$	9	123	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
VP_{15}	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1	40	132
VP_{16}	-	-	-	-	-	-	-	-	122	37	0	2	-	-	-	-	-	-	-	-	-	-	-	-	-
VP_{17}	7	132	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
VP_{18}	129	24	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
VP_{19}	1670	2284	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
VP_{20}	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1195	2759	-	-	-	-	-	-	-	-	-
VP_{21}	452	3502	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
VP_{22}	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	5	364	17	66	3502	-	-	-	-

the system as a variation point or if it should be considered as a customized development for a given customer.

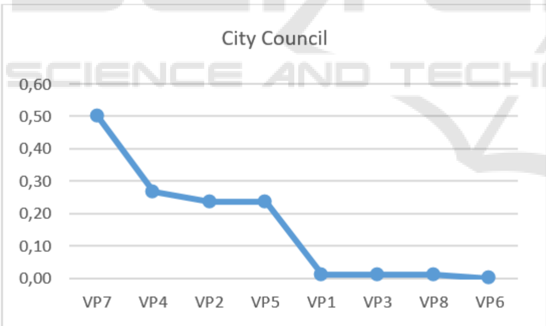

Figure 1. Use of the variation points for councils.

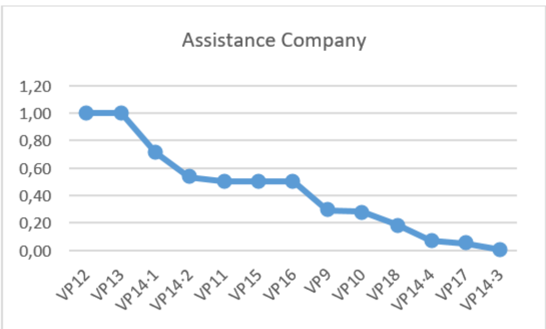

Figure 2. Use of the variation points for companies.

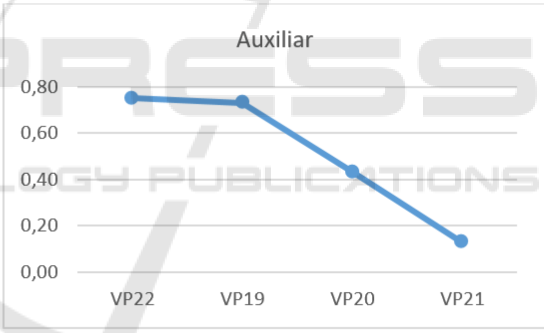

Figure 3. Use of the variation points for auxiliar assistance workers.

5 CONCLUSIONS

In this paper, we have presented a real experience on using variability management to analyze and model the variability in a complex real web system delivered as a service. Applying variability management has allowed the developers to analyze the variation points of the system under a solid framework, and in a systematic and comprehensive way. This approach has allowed the development team to deal with variation in an easy and natural way.

In addition to presenting the system and the variation points we have identified during its development, we have presented real data on the deployment of the system and the use of the variation points introduced in the system by the different user profiles it supports (councils, companies, and auxiliar assistants). In this analysis we have found interesting results regarding the use of the variation points. Some of them are considered by the company that developed the system as worthwhile or profitable, since many customers use of the options and also many users use the other options for that variation point. However, the data also reflect that some variation points have been developed only for the specific need of one specific customer, so they are considered by the company that developed the system as less profitable.

In addition, while dependencies had not been found during the variability analysis, the data suggests that some dependencies do really exist among different features, since the number of customers using one of its options is the same.

We consider that these results raise interesting questions for future work, in which we are currently working. First, some way of introducing the concept of return on investment in the variability management framework would be of interest for many companies, since this would help not only in identifying and analyzing variation points, but also in making the decision of whether introducing them in the system is worth or not. On the other hand, this real project has helped us to detect that identifying dependencies between features is not always direct during the analysis phase. Analyzing the real-world data collected from real systems to try to automatically identify such dependencies is also an interesting problem for future work.

ACKNOWLEDGEMENTS

This work has been funded by Xunta de Galicia / FEDER-UE CSI: ED431G/01 (Centros ingulares de Galicia), Xunta de Galicia / FEDER-UE CSI: ED431C 2017/58 (Grupo de Referencia Competitiva), MINECO-CDTI / FEDER-UE CIEN LPS-BIGGER: IDI-20141259, MINECO-CDTI / FEDER-UE INNTERCONECTA uForest: ITC-20161074, MINECO-AEI / FEDER-UE ETOME-RDFD3: TIN2015-69951-R.

REFERENCES

Apel, S., Batory, D., Kästner, C., Saake, G., 2013, "Feature-Oriented Software Product Lines: Concepts and Implementation", Springer.

Siegmund, N., Sobernig, S., Apel, S., 2017, “Attributed variability models: outside the comfort zone”. *In Procs. of the 11th Joing meeting on Foundations of Software Engineering (ESEC/FSE 2017)*, ACM, pp. 268-278.

Moens, H., Truyen, E., Walraven, S., Joosen, W., Dhoedt, B., and De Turck, F, 2012, “Developing and Managing Customizable Software as a Service Using Feature Model Conversion,” *in Procs. of the IEEE Network Operations and Management Symposium (NOMS 2012)*, pp. 1295–1302.

Lizhen, C., Haiyang, W., Lin, J. and Pu, H., 2010, “Customization Modeling based on Metagraph for Multi-Tenant Applications,” *in Procs. of the 5th International Conference on Pervasive Computing and Applications (ICPCA 2010)*, pp. 255–260.

Mietzner, R., Metzger, A., Leymann, F., and Pohl, K., 2009, “Variability Modeling to Support Customization and Deployment of Multi-Tenant-Aware Software as a Service Applications,” *in Procs. of ICSE Workshop on Principles of Engineering Service Oriented Systems (PESOS 2009)*, pp. 18–25.

Weiping, L., 2009, “An analysis of new features for workflow system in the SaaS software”, *in Procs. of the 2nd Int. Conference on interaction sciences: information technology, culture and human (ICIS 2009)*. ACM Press.

Galster, M., Zdun, U.,Weyns, D., Rabiser, R., Zhang, B., Goedicke, M., Perrouin, G., 2016, “Variability and complexity in software design: towards a research agenda”. *ACM SIGSOFT Software Engineering Notes*, 41 (6), ACM.