

A Toolkit Method to Match Up End User Needs with Salesforce.com Solutions

Ken Decreus¹, Stijn Viaene², Geert Poels¹

¹ Faculty of Economics and Business Administration, Ghent University, Belgium.

{ken.decreus | geert.poels}@ugent.be

² Vlerick Leuven Gent Management School, Vlamingenstraat 83, 3000 Leuven, Belgium

stijn.viaene@vlerick.be

Abstract. The recent trend in Software-as-a-Service (SaaS) offers the end user ready-to-use software systems via a new delivery model. Market leader Salesforce.com can be seen as the prototypical implementation of SaaS software. One of the key assumptions made by Salesforce.com is that end users do not need explicit support to express their user need information. Therefore, no explicit methodology is offered to combine solution information with the end user's need information to design a responsive Salesforce.com product. We will provide a method for end users to express their requirements in order to discover how the Salesforce.com Sales Force Automation (SFA) application should be configured. By using the toolkit, the end user will discover which part of his needs are covered by means of the SFA application. This way, the user will be able to start his end-user development path for the covered functionality, and use the resulting toolkit models to communicate to other stakeholders what is missing in the Salesforce.com SFA application.

Introduction

The recent trend in Software-as-a-Service (SaaS) offers the end user ready-to-use software systems via a new delivery model. Market leader Salesforce.com can be

seen as the prototypical implementation of SaaS software. One of the key assumptions made by Salesforce.com is that end users do not need explicit support to express their user need information. Therefore, no explicit methodology is offered to combine solution information with the end user's need information to design a responsive Salesforce.com product. We will provide a method for end users to express their requirements in order to discover how the Salesforce.com Sales Force Automation (SFA) application should be configured. By using the toolkit, the end user will discover which part of his needs are covered by means of the SFA application. This way, the user will be able to start his end-user development path for the covered functionality, and use the resulting toolkit models to communicate to other stakeholders what is missing in the Salesforce.com SFA application.

1 Method

1.1 Example Organisational Setting

The sales department of a fictive enterprise in the telecommunications sector called TechCom incorporates a lot of best-practices known, as the sales manager used to work for one of the big consultancy offices. The department is currently led by the sales manager, who reports to the vice-president of Mid-Markets. Two sales teams are working for the sales manager, each team consisting of five account managers. The account managers are the single points of contacts to the account, looking for opportunities to sell products and services and trying to build good relationships with the contacts at the account. When TechCom organizes a big annual event, they ask visitors who want more information to fill in a feedback form. These possible new customers are given to the Sales Manager, who keeps an Excel sheet with possible leads and customers-to-be-contacted. During the weekly conference call with all account managers, the sales manager assigns the new leads to the relevant account managers. As the sales manager has little time to double check these assignments, sometimes leads are getting lost. Furthermore, the sales manager would like to impose a standardised way to qualify a lead, because he suspects that some of his account managers convert leads into contacts too soon to reach their monthly targets.

The sales manager decided that he would save time by automating the current sales processes by means of Salesforce.com SFA, but he feared that the current timing was badly chosen. First of all, the VP Mid-Markets has a deep distrust of hosting sensible customer data at a vendor site. Therefore, he instructed the sales manager to keep the financial customer data locally in the ERP system, and wanted to have a secure bridge between the local and the external SaaS server. Secondly, because the sales manager wants to gain the trust of his boss, he

decided to limit the project scope to the lead management processes; when trust has been established, the sales manager would like to extend the automation scope to opportunity management and account/contact management. Thirdly, the sales manager wants to find a consensus on the requirements needed with his colleagues from other company sites, as sales managers from other country sites are instructed –when implementation was successful– to work on the same system. Finally, due to an unfortunate coincidence, the wife of the sales manager is pregnant and the manager was granted a parental leave for three months after this SaaS automation project. A senior consultant will fill up this gap, but only a few days of knowledge transfer are foreseen.

1.2 Toolkit Method for End User

Research in the area of Management Science suggests that product design done by the product user is far more efficient than innovation by product manufacturers [1]. It is proposed to outsource need-related innovation tasks to the users themselves after equipping them with *toolkits* for user innovation. As an example in the business/IT field, Ricken & Steinhorst [2] propose to empower a business user by considering the Supply-Chain Operations Reference-model (SCOR) as a toolkit for business process innovation. In our research, we propose to use the first phases of the Tropos methodology [3] as a toolkit mechanism. The Tropos project provides a model-driven methodology where *i** models [4] are used to drive the generation of software systems.

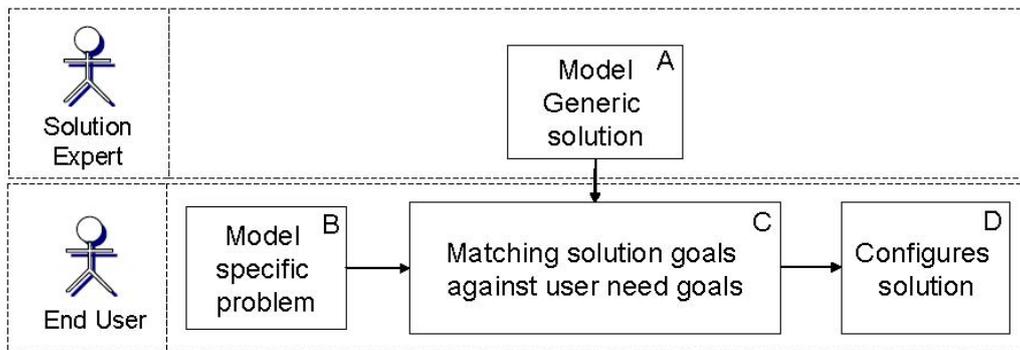


Figure 1. Toolkit method for end user.

We propose a toolkit method for end users, as displayed in Figure 1, where a solution expert creates a model of the generic solution (Part A) and the end user models his specific problem (Part B). Later on, the end user matches both models in order to understand where the solution supports his needs (Part C) and configures the fractions of the solution that supports his needs (Part D).

The *i** modelling framework provides us with different modelling constructs to specify intentionality. A *goal* node in the goal tree shows that there are alternative ways of achieving the goal, but *no specific instructions* are given how to achieve

the goal (e.g. when a car owner enters a repair shop and asks to “just get it fixed”). A *task* node shows that we *specifically* know what to do but there are *constraints* on how to do it (e.g. the car owner asks the repair shop to raise the engine idle settings in order to fix the engine).

Given that the generic solution has been modelled in *i** by an application expert (Figure 2 – Part A), the end user can import this solution model into the toolkit environment. Applied to the TechCom case study, the sales manager models his specific problem (Figure 2 – Part B) using the *i** modelling language.

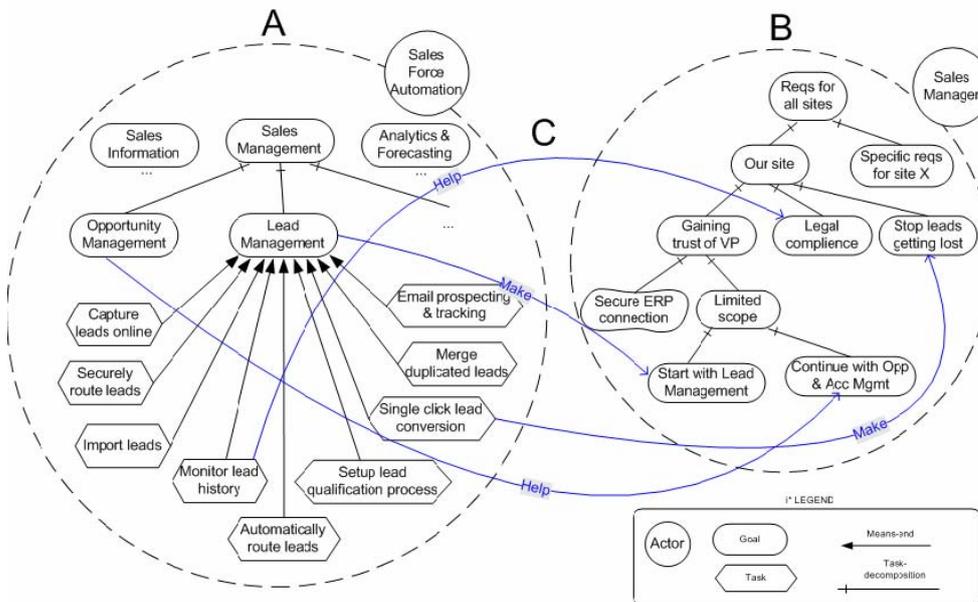


Figure 2. Matching generic solution model with specific problem model

By drawing *contribution relationships* between the relevant goals, the end user is able to express the degree to which the SaaS system supports his goals (Figure 2 – Part C). Note that automated support is needed for helping the user to know which arrows to draw. In the context of non-functional requirements, Castro et al. [3] propose the contribution relationships *help* (partial positive), *make* (sufficient positive), *hurt* (partial negative) and *break* (sufficient negative). Nevertheless, these contribution relationships could also be seen in a more general context [5] where contributions are expressed between both functional and non-functional requirement goals. After matching the generic solution model with the specific problem model, we obtain covered requirements (Figure 3 – Covered zone) and requirements that are not supported by the solutions’ capabilities (Figure 3 – Problem zone).

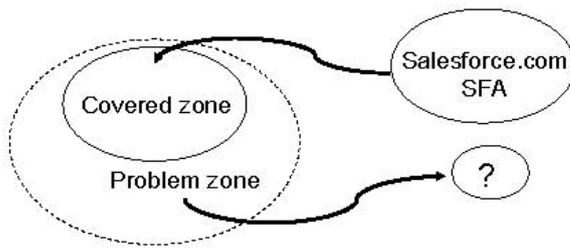


Figure 3. The SFA application may not cover all requirements

In the covered zone, the use case ‘Monitor lead history’ *helps* to enforce the legal compliance, while use case ‘Single click lead conversion’ *makes* the prevention of leads getting lost. Limiting the scope of the SFA application to the lead management module *makes* the establishment of further trust. The problem zone shows the limitations of the SFA application: no secure ERP connection is foreseen, opportunity and account management is planned to install on medium-to long-term, still further trust of VP is needed and legal compliance is not fully supported by a SFA use case.

Finally, guided by the covered requirements, the end user follows the instructions of the SaaS documentation to install the solution that supports these requirements. For instance, configuring the ‘Single click lead conversion’ use case is fully specified in Salesforce.com end user documentation (see Figure 4).



GETTING THE MOST FROM YOUR LEADS

Leads help you track your potential business. Using leads gives you a streamlined way to start your business flow from potential customer to closed deal.

Converting Your Leads

As soon as you qualify a lead, you'll want to convert it into an account, contact, and opportunity so you can start to close the deal.

1. Click **Convert** on the Lead detail page.
2. Verify the owner of the new records and optionally send him or her an automated notification email; set the status of the converted lead; schedule a follow-up task; and click **Convert**. We try to match the new account and contact to existing accounts and contacts if possible. You can choose to use the existing items or create new ones.
3. Activities related to the lead are now associated with the newly created account, contact, and opportunity.
4. Converted leads can no longer be viewed in the Leads tab but they will contribute data to reports. When customizing your report, enter a filter option of "Converted equals True" to view converted leads.

- The **Company Name** from the lead becomes the **Account Name**.
- The **Lead Name** from the lead becomes the **Contact Name**.
- The opportunity and contact you generate are associated with the account.
- If enabled for your organization, universally required custom fields, workflow rules, validation rules, and Apex triggers are all enforced when you convert a lead.

How is a Lead Different from an Account, Contact, or Opportunity?

- An account is the company information for a deal.
- A contact is an individual associated with an account.
- An opportunity is the related deal information.
- Leads can be automatically converted to accounts, contacts, and opportunities.
- Accounts, contacts, and opportunities present a total picture of a customer relationship. A lead is the introduction to that picture.

Figure 4. End User Documentation for ‘Single click lead conversion’ use case

2 Conclusions

Triggered by the fact that Salesforce.com does not provide explicit RE support to end users, we believe that end users provided with RE support could obtain fast time-to-market of SaaS applications. This paper proposed a requirements specification method to allow end user to express their problems in order to select the correct Salesforce.com SFA functionality. Future work will validate our contribution and show the generic applicability of our method.

3 References

1. von Hippel, E., Katz, R.: Shifting Innovation to Users via Toolkits. *Management Science* **48** (2002) 821-833
2. Ricken, A., Steinhorst, A.: Why working with reference models increases process innovation. *BPTrends* **February** (2006)
3. Castro, J., Kolp, M., Mylopoulos, J.: Towards requirements-driven information systems engineering: the Tropos project. *Information Systems* **27** (2002) 365-389
4. Yu, E.S.-K.: Modelling strategic relationships for process reengineering. University of Toronto (1995) 181
5. Markovic, I., Kowalkiewicz, M.: Linking Business Goals to Process Models in Semantic Business Process Modeling (to appear). *EDOC Proceedings* (2008)