BPMN AS A BASE FOR CALCULATING THE TARGET VALUE OF EMPLOYEES' SECURITY LEVEL

Jan Schlüter[1], Stephanie Teufel[2]

[1,2]**iimt, University of Fribourg, Switzerland**

[1]**jan.schlueter@unifr.ch,** [2]**stephanie.teufel@unifr.ch**

ABSTRACT

In this paper we aim to create a new model which uses the Business Process Modelling Notation (BPMN) as the base for calculating the target value of the employees' security level. It has to be assumed that all processes, at least those which interact with the information technology directly, are written down in BPMN or a fully-convertible notation respectively describing language. It is important that bigger companies and public authorities do fulfil this requirement.

The problem is that it is hard to get an overview about the information system access and privileges of each employee in bigger companies. The approach currently used in "secaliser", our initial project, expects that the information technology affinity is based on each employee's job position. In many real life situations this condition cannot be fulfilled. Due to the fact that we try to optimise the trainings in an economic way, there is only a small range between the necessary and the optimised security level. This is the reason why it is so important to enhance the exactness of our calculation. The goal of the described model is to make conclusions concerning the specific should-be security level of each employee, based on comprehensible data, which is extracted from the company processes.

KEY WORDS

BPMN, security level analysis, employee, process analysis

BPMN AS A BASE FOR CALCULATING THE TARGET VALUE OF EMPLOYEES' SECURITY LEVEL

1 INTRODUCTION

Nowadays, business becomes more and more complex. The same applies to the structure of the information systems, because they try to copy a model of the complex business. Of course, many of these complexity problems do not affect the information security directly, though there are some security relevant factors which increase disproportionately high with growing business complexity. [8]

In the past, there were many technical ways on how to protect information from being accessible, to be changed, or to be taken away, but those systems do not help us to go the first step: finding out who needs which permissions and how to structure the way the access permissions are allocated. Regarding a suggestive restructuring of business processes in order not to give away the same permissions to a larger than necessary group of employees, is an absolutely important step that is completely untended. [2]

It must not be disregarded that there are different kinds of information to manage. Some information is stored in file systems and the access to the different shares is limited to authorised users. Due to the fact that those mechanisms are implemented in most common operating systems, they are well-known – as well by end users as system administrators – and caused by the simple permission structure these kinds of information can be easily managed, and the access permissions can be clearly arranged. Much more complex in managing the access are other systems which have, in most cases, not such a clear structured base, this also afflicts plain inheritances like normal folder structures do. Although relational storage models can also be managed in a descriptive way according to tables – even if the access permissions can be allocated here much more sophisticatedly – role based access models are not as widespread as in file systems. However, there are other platforms which cannot be managed as well in a descriptive way as relational and folder based structures, for example, hardly adaptable and closed third party software. Beside the fact that the company cannot guarantee the permission system inside the software, it could be hard to find out the kind of information each employee has contact with. Due to the fact that big-

ger software systems often directly use the permission system of a relational database management system, this problem affects smaller software products much more than the bigger ones. Therefore many niche software products are affected, but these software programs may contain the most critical data for the business. [2]

2 PROCEEDING

Due to the fact that not all information is security sensitive on the same level, it becomes important to group the different kinds of information with same security levels and rate those groups. These information groups are the initial point for our considerations: In an internal feasibility study, we checked out different ways on how to get information of the employee's specific security affinity, and the security level we should use for the optimisation. An approach which uses the Business Process Modelling Notation (BPMN) to check all data the employee has contact with, is able to change, or delete, is the most promising one. In addition to the comparatively simple as-is state analysis, this approach gives the possibility to monitor changes in the different processes and to react on those changes contemporarily. Due to the power of BPMN, especially the artefact-components, there are several ways to bind processes, persons, and data-permissions with each other, without breaching the current working draft from 3rd May 2004. [5, 3]

2.1 Grouping information

To get an overview of the permissions an information (or data) directly or indirectly implicates, it is important to collect all available information. The groups which should be created, as described before, have to be divided into two different groups, namely: *Security Sensitive* and *Data Equivalency/Implication*.

Security Sensitive

It makes sense to create groups and directly link them to a security sensitive level. This level is a simple number and represents the security importance of the grouped information. Furthermore, due to gaps between the security sensitive levels of the different groups, it may be necessary to devide critical and non-critical information. Anyhow, it will be hard to scale the groups

in a fair way, because the different information might be very hard to compare with each other, but this problem has to be solved by the company themselves.

Data Equivalence/Implication

That there are different information in the same group does not declare the information as equivalent, because completely different information which may belong together or not, can be on the same security level. For example, *street* and *zip code* as part of an address data set, or two non-correlating information like a *social security number* and an *image*. The reason why we do not only use equivalence groups and link them – of course not unique – to the security sensitivity level is self-explanatory when trying to build up the first business process model: Indeed, equivalent information or information which implicate other information will normally be in the same group, but also, in this opposite reflection, the statement is not universally valid. In our own tests, we exposed that it might be useful to adapt the view in some cases and to divide own personal information from the personal information of a third party.

To mark information as privately accessible is particularly suitable for every kind of identification of the person which is not done automatically, for example, at the cashier's desk where the banker does not ask you for any information of your banking account because they personally knows the (manually) identified person. To describe these social problems with business process models only would be very weak in practice.

As not to breach the BPMN standard, we decided to do the distinction between the two kinds of personal information very simply and just appended a wildcard to the information name which is explicit, not being accessed by any other than the belonging person. This simple approach does only influence the naming convention and not the standard itself.

2.2 Grouping employees

In order to make it possible to point out the employees who do have more permissions than other employees in the same position, it is useful to also group those persons who can be compared with each other. A comparison may be useful with either employees in the same job position or with those who are located at the same place. The employee groups will only be used

to display the result better and to highlight some irregularities. Neither the calculation nor the results are affected by these groups.

3 CREATE A BUSINESS PROCESS MODEL

As written in section 1, the business process model should be created using the BPMN standard. When analysing the different business process model standards, it came out that the functional range of BPMN is much wider than others like the event driven process chains (EPC). Due to the fact that the BPMN standard is administered by the Object Management Group since 2005, and due to their experiences with the Unified Modelling Language (UML), it can be expected that the BPMN functional range for information systems can be increased in the near future. [7, 6, 5]

Normally, bigger companies and public authorities already have at least some of their business processes written down. Of course it will not make sense to create hundreds of processes again that are already written down, but due to the complexity of the BPMN standard it is assumed that it is possible to import other business process formats into BPMN with no or a very small information loss. Common used process description standards like EPC as part of the ARIS Framework do have such a little modelling complexity that the transformation can be done without any information loss (all elements of EPCs are also part of the BPMN standard). Therefore it is only required to add additional information to the already existing processes. [10, 4] Of course it is not realistic for every branch that all processes are already written down, but especially for those processes which concern security sensitive information the assumption is reasonable.

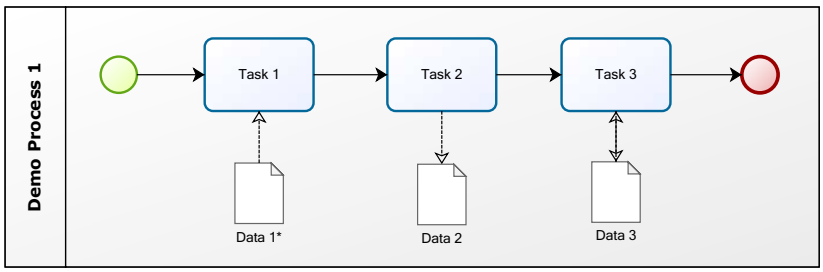

Figure 1: Demo Process 1

Figure 1 shows a demo process created in BPMN containing three *Tasks* and three *Data Objects* in one *Pool*. Except for the wildcard appended to *Task 1*, this is a sample business process which can be found in companies (for example in most applications that manage data permissions like CMS systems), although many companies are using simple EPCs until today[1]. Analysing this process could proceed straight forward, because there are no conditions or anything else to combine with. Even the three Data Objects are not connected through any other *Task* or *Message Flow*. As described in 2.1, the wildcard is normally used in *User Tasks* only, which are manually performed. Inexactnesses like this, which do not need to be caused by the company itself, but maybe by the use of weak tools or export mechanisms will make it difficult to get an overview about the business processes. Therefore it is absolutely necessary that after importing into or creating business processes, the complete process has to be rechecked for a consistent and correct use of BPMN elements and our adapted naming convention. The following description gives a short summary about what happens in *Demo Process 1* for those who are not familiar with BPMN:

- The *Process Sequence Flow* starts, initiates three *Tasks* and ends.
- *Task 1* seems to be a *User Task*, because only the owner of the data (remember the wildcard) is able to read (incoming arrow) the *Data Object*.
- *Task 2* is a *Task* which directly writes (outgoing arrow) into the *Data Object*.
- *Task 3* is a *Task* which reads and writes (incoming and outgoing arrow) the *Data Object*. Typical use of this read and write actions are normally conditional updates or the use in *Sub Processes*[2].

The way how the processes should be created is free to the company, but of course it is useful to use one of the well-established standards for all processes. Due to the fact that the process in detail is much more interesting for the analysis (very general business process parts almost contain no data information), we recommend to use a bottom-up approach. This course of action will ensure that not all data sets are defined from the beginning

[1] against definition, our demo process does neither start nor finish with an event
[2] a non-atomic process

on, but that most of the defined assignments are already meaningful while the business process is not complete. This could lead to a first rough restructuring of the business models while still modelling business processes. [7]

3.1 A sample process for analysing

The problem in Figure 1 was the presumable non exactness of the used BPMN items.

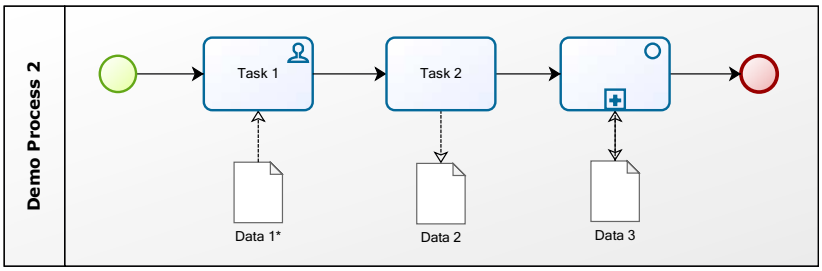

Figure 2: Demo Process 2

The *Demo Process 2* in Figure 2 corrects this problems and now represents a good base for further adaptions. Neither the *Sequence Flow* nor the *Tasks* respective *Sub Processes* have changed their intent until now, or will change it in the following steps.

In the following step we group our *Data Objects*. Figure 3 shows the result of two groups being created:

- *Data 1* and *Data 3* are implicating each other. This means that the *Data Objects* themselves are not equivalent, but that a change of *Data 1* induces a change of *Data 2* and conversely.
- *Data 1* and *Data 2* are assigned to *security level 2*, which means that the *Data Objects* contain security sensitive information.

Note: Even if *Data 1* and *3* are implicating each other this does not mean that those *Data Objects* have to be security sensitive on the same level in general. *Implicating Data Objects* are related to each other and this relation does not need to be connected directly across one object. Also

objects (in context of business model representation), can relate to each other and their attributes may be transitively connected to each other, which by the way may cause the relation between the objects. Comparing this general model with relational database management systems, in combination with data consistency systems as used in enterprise programming, for example Hibernate and Java, will help us to understand the coherences. Due to complexity reasons, our example does not contain any *Message Flows* which would necessarily be building a real business process model, but is uncared for in our analysis. [1]

Considering Figure 3, it will become clear that it will be impossible to either visualise all equivalent and implicating *Data Objects* as one visual group, or to group the different security relevant *Data Objects* to only one unique visual group with the same security level. With only three Data Objects and two Groups, the Diagram has to tend towards to the bottom in order to visualize that *Data 2* is not part of the *Implicating Data Group*. Due to this, there are three different ways:

1. Keep it like it is and only very simple group configurations can be set. This approach would not need any further adaptions, but will lead to a point where you cannot model your business processes in an accurate way, and moreover, the business developing process would take some more time, because doing the layout of the *Data Objects* will become hard.

2. Keep the editor like it is today and just name belonging groups identically. So it could happen that there are multiple groups with the same security level containing different *Data Objects*. This approach would be easy to implement, but cause a loss of the general diagram overview at more complex diagrams.

3. Completely rebuild the editor and integrate different views. In the default mode, every *Data Object* is a member of the different groups, which are listed below each object, and a second view mechanism waits for one special group being selected and arranges all items in a way that they can be displayed as a group. This approach is much more complex and causes to a loose of controlling the business process layout. Also this implementation is much more difficult than the other two options.

Due to the fact that we try to model the business processes of bigger companies, the first option does not fit to our mission. The second, quite

simple to adapt approach should be implemented anyway, because this is the simplest suitable solution and the only one which gives the process creator the possibility to layout everything manually[3]. The third option is the most beneficial one, but doing some tests in automatically arranging and doing the layout of the different groups caused some problems. A problem we cannot currently solve suitably is to arrange groups across different *Pools* and *Lanes*, while we have no problem with *Sub Processes*, which can be, in contrast to the *Pools* and *Lanes*, easily expanded.

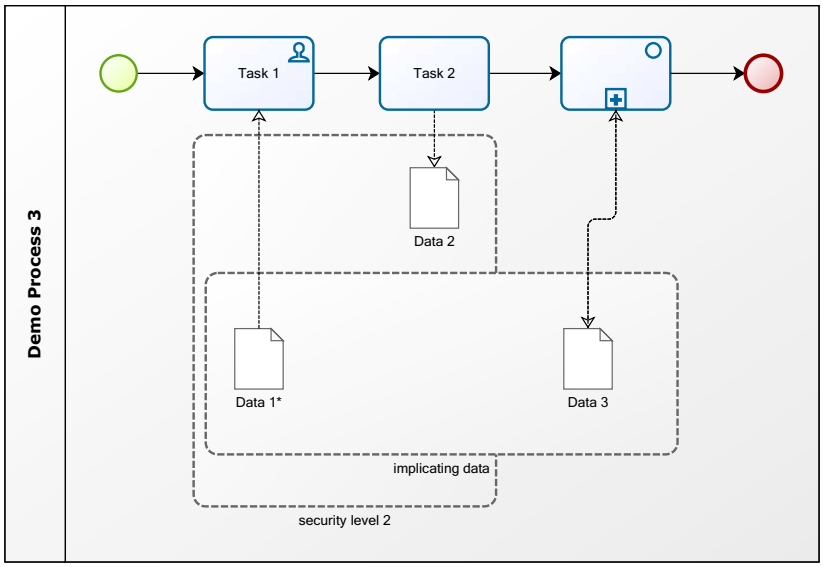

Figure 3: Demo Process 3

Applying a view only for grouping *security level 2* is shown in Figure 4. Of course the example process is very simple and even without focussing on the one group, the process was clearly arranged before, but this would change in bigger processes and it has to be noted that it is not possible to visualise all groups at the same time.

[3]it would almost be impossible to layout every view of option 3 manually, because you have to update everything after just a minor change

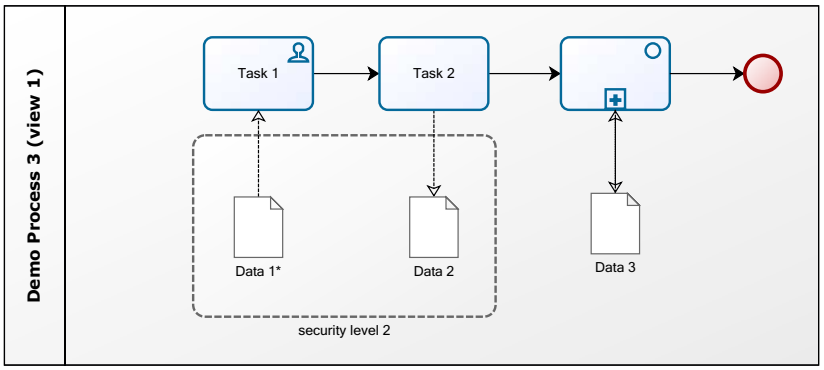

Figure 4: Demo Process 3 (view 1)

3.2 Assigning tasks

BPMN offers the possibility to assign tasks with either *Entities* or to *Roles*, where *Roles* can be either employees, job positions, or other identifying attributes to represent one person or a group of persons. Unfortunately these assignments are hidden attributes of the BPMN standard and will not be printed out, but it is arguable whether additional information in the diagram would make the diagram really more informative or just confusing. However, every *Task* can be linked to a *Role* and due to this we get a direct link from the employee to the necessary user permissions.

Like managing file permissions, it is recommended not to use a certain employee to assign user permissions, but to use user groups[4]. Comparing the employees who are members in the different groups is from the technical point of view very simple, but allows the management to find out which employee has more or less user permissions in comparison to other employees in the same job position.

4 SECURITY LEVEL

The outcome of the described approach will give us information about each employee's security level. This security level is distinct for every employee, and is not mandatorily based on job positions or any other common infor-

[4]these groups have nothing to do with those in 3.1

mation which is currently be used in companies. Especially when planning trainings for employees, it is very important that the distribution of the available resources makes sense. Dedicating too few resources for certain employees could result in security leaks, too many resources could waste resources which will in common result in too few resources for the more important employees. To find the right balance is very hard. [9]

4.1 Procedure

Before starting, we remember our two groups from 2.1 namely *Security Sensitivity* and *Data Equivalence/Implication*. Because it does not make any sense to factor multiple equivalent or implicating security sensitive data objects twice, we will have to find the highest rated security sensitive information which is accessible by each employee per *Data Equivalent/Implicating* group.

Table 1 will give an example were the *Level* column describes a *security sensitivity* group and *Equivalence* a *data equivalence/implicating* group.

Level 1	Level 2	Level 3	Equivalence 1	Equivalence 2
a	b	c	a	d
d	e	f	c	b
g	h	i	e	
j	k	l		

Table 1: Group Table

First of all we need to mention that every task (a to l) is accessible by the sample employee. Second, we see that we can delete every task in our table which has a more security sensitive task in the same equivalence group. Having a look at the highest security level of each equivalency group in Table 1 will show that the tasks a, c and d are redundant because each of those have a more security sensitive task in their equivalent group. The result of this simplification is shown in Table 2.

As shown in Table 2 there are nine security sensitive tasks left. Those tasks can be accumulated which result in a target value of security level of 19.

Level 1	Level 2	Level 3
~~a~~	b	~~c~~
~~d~~	e	f
g	h	i
j	k	l

Table 2: Security Table

5 PROBLEMS

It will be hard for the company to cover the complete business with well defined business processes. The initial effort is very high and increases with the complexity of the business, or in other words: with the number of items in the business process.

Furthermore, even this approach will lose clarity when there are too many groups, or the employees are too far-scattered into these groups so that there are no patterns to identify manually. However, the final step, namely the analysis and evaluation of the results, has to be done manually.

In addition to the described features, it would be nice to help analyse the results and to recommend ways on how to change the business processes to increase security, but this feature can only be developed after having some companies which have produced sample process data which can be analysed manually. These general ideas of improvement have to be transferred into automatic algorithms.

6 CONCLUSION

Business Process Models as being used in modern companies can be used for much more than only to display and analyse workflows or for the accreditation of the business model.

First of all, by means of the company's business processes, it can be detected automatically which user permissions an employee needs to have to do this work correctly. This approach works across the boundaries of isolated information systems and gives a detailed overview over all accessible information inside the company. Worthy of mention is that the model uses the real company view and not a model being distorted by information permission systems. However, transferring the user permission information automatically into the individual permission systems should be possible with most

information systems. The other systems could be set up manually whenever the access to the corresponding information changes. Of course this approach does not help directly to improve the permission systems of the information systems, but it shows up which different roles should be available with which specific user permissions. Furthermore, the problems of the companies current permission systems will come, out and it is possible to point out where the detailed problems in the allocation of user permissions occur and which software has to be adapted to make it possible to protect information that is not necessary to access for certain groups, but security sensitive.

In a second step, we can use this detailed information to get information about the target value of employees' security level. As described in [9], it is absolutely necessary to have information about each employee's security affinity in order to plan company trainings correctly and to think about restructuring either certain parts of the business processes or of the company's organisation structure.

In bigger companies it is not that there is no information which concerns the security level of an employee, but that this information is included in business processes which are at present insufficiently used.

References

[1] Bauer, C., and King, G. *Java Persistence with Hibernate*, revised ed. Manning Publications, 2006. ISBN 978-1932394887.

[2] Benantar, M. *Access Control Systems: Security, Identity Management and Trust Models*, 1st ed. Springer, 2005. ISBN 978-0387004457.

[3] Jeston, J., and Nelis, J. *Business Process Management, Second Edition: Practical Guidelines to Successful Implementations*, 2nd ed. Butterworth-Heinemann, 2008. ISBN 978-0750686563.

[4] Madison, D. *Process Mapping, Process Improvement and Process Management*, 1st ed. Paton Press, 2005. ISBN 978-1932828047.

[5] Object Management Group. `http://www.omg.org/spec/BPMN/1.1`. Website, 2008. last visited: 25.4.2008.

[6] Object Management Group. `http://www.omg.org/spec/UML/2.1.2`. Website, 2008. last visited: 25.4.2008.

[7] Scheer, A.-W., Kruppke, H., Jost, W., and Kindermann, H., Eds. *Agilität durch ARIS Geschäftsprozessmanagement: Jahrbuch Business Process Excellence 2006/2007*, 1st ed. Springer, 2006. ISBN 978-3540333586.

[8] Schlüter, J., Novy, B., Teufel, S., and Marx-Gomez, J. Automatisierte erstellung von wissensbilanzen. In *Multikonferenz Wirtschaftsinformatik 2008* (2008), M. Bichler, T. Hess, H. Krcmar, U. Lechner, F. Matthes, A. Picot, B. Speitkamp, and P. Wolf, Eds., GITO-Verlag. ISBN 978-3-940019-34-9.

[9] Schlüter, J., and Teufel, S. secalyser - a system to plan training for employees. In *Proceedings of the International Symposium on Human Aspects of Information Security & Assurance (HAISA 2008)* (2008), S. Furnell and N. Clarke, Eds., vol. 2. to be published.

[10] Weske, M. *Business Process Management: Concepts, Languages, Architectures*, 1st ed. Springer, 2007. ISBN 978-3540735212.