



universität  
wien

# Masterarbeit

Titel der Masterarbeit

Analyse of Critical Factors for  
Business Process View merging

verfasst von

Claudia Gratz, BSc

angestrebter akademischer Titel

Diplom-Ingenieurin (Dipl.-Ing.)

Wien, 2015

Studienkennzahl lt. Studienblatt:  
Studienrichtung lt. Studienblatt:  
Betreut von:

A 066 926  
Masterstudium Wirtschaftsinformatik  
Univ.-Prof. Dipl.-Math. Dr. Stefanie Rinderle-Ma



## Acknowledgement

At first I would like to give big thanks to the Raiffeisenlandesbank Niederösterreich-Wien which provided me with all the information I needed for the procured process. Special thanks go to all my colleagues from the department MRM whom I surely annoyed with my many, many questions.

I would also like to thank my supervisor Rinderle-Ma Stefanie who encouraged me for my idea for this master thesis. Further thanks go to her for sparking the great discussion which brought many aspects of the topic to me I wouldn't have thought about myself. Thank you for the great continuous support.

Special thanks go to my old friend Safranek Stefan who helped me out by proofreading everything I wrote. It surely was a great pain to correct my bad English. Thank you for all the hours you spent to help me out.

Last but not least I'd like to thank my family which supported me in this hard and long time. You always helped me to encourage me when I got stuck in writer's block and also pressured me to work when the weather was nice and I wouldn't want to work. Thanks for the great support not only for the master thesis but for my whole studies. Without you guys I couldn't have made it.



## Contents

1	Introduction .....	7
2	Research Method .....	8
2.1	Conceptual Model of this master thesis .....	10
2.1.1	Criteria.....	12
2.1.2	Factors.....	12
2.1.3	Rating .....	13
2.1.4	View.....	13
2.1.5	Person.....	13
2.1.6	Level .....	13
2.2	Further examples and evaluation .....	14
2.3	Introduction of the used tool.....	14
2.4	Summary about the research method .....	15
3	Factors.....	16
3.1	Identifying and Selection of factors .....	16
3.1.1	Analysis of the literature .....	16
3.1.2	Analyse of the used tool ADONISCE2.....	16
3.1.3	Evaluation of the potential factors.....	18
3.1.4	Summary of the factors.....	31
3.2	Summary of the Analyse .....	31
3.3	Point System.....	33
4	Ranking system and Merge .....	33
4.1	Mathematic formula .....	34
4.1.1	Result: mathematic formula.....	37
4.2	Mathematic formula with random factor .....	38
4.2.1	Result multiplied with random number.....	38
4.2.2	Individual factors multiplied with random number .....	41
4.2.3	Summary over the mathematical approaches.....	42
4.3	Developed Algorithm .....	42
4.3.1	Description of the developed algorithm .....	44
4.3.2	Summary of the developed algorithm .....	50
5	Testing on case study .....	52
5.1	Testing case Prolima control .....	52

5.1.1	Introduction of the process.....	52
5.1.2	Department scenario .....	53
5.1.3	Individual views and the positions in the scenario .....	54
5.1.4	Merged model.....	58
5.2	Testing case university .....	61
5.2.1	Introduction of the process.....	61
5.2.2	Department scenario .....	63
5.2.3	Individual views and the positions in the scenario .....	64
5.2.4	Merged model.....	66
5.3	Testing case KFZ .....	68
5.3.1	Introduction of the process.....	68
5.3.2	Department scenario .....	69
5.3.3	Individual views and the positions in the scenario .....	70
5.3.4	Merged model.....	74
5.4	Summary of the testing.....	76
6	Evaluation of the testing .....	77
6.1	Evaluation on equality of activities .....	77
6.2	Evaluation of the structural differences.....	79
6.3	Summary of the Evaluations .....	80
7	Discussion.....	81
8	Future work.....	82
9	Related work .....	83
10	Summary .....	85
11	References.....	86
12	Appendix .....	88
12.1	Interview guideline (German) .....	88
12.2	Prolima control process .....	89
12.3	University noting process.....	105
12.4	KFZ insurance process .....	115
12.5	Abstract .....	134
12.5.1	English .....	134
12.5.2	German.....	134
12.6	Zusammenfassung.....	135
12.7	Curriculum Vitae.....	136

## 1 Introduction

Business process modelling (BPM) is a highly recommended task for each company. [1] Companies across the whole world are using BPM to get a clear understanding of processes as well as a view on tasks performed in processes. To model these processes can be a tough problem [2]. This is because the word modelling is recognised and used worldwide but most people who execute the modelling are rather new to this field and therefore the results of their work are often not a hundred percent correct. [2] Even if the modeller has enough knowledge of the field the difficulty of how to get enough and correct information is still rather high. When the processes are documented for the first time or adaptations should be made because optimisation has to be done the information gathering is a difficult part. Information is researched via different methods like questionnaires but this is also rather difficult because the right people have to be found and the correct questions have to be asked. [3] Even if this is all accomplished it can't be guaranteed that the answers are usable. Nevertheless, if this hurdle can be taken the next problem is even bigger because now information is known but the next problem is to get a process out of it. The most time is needed to filter the different answers and put them together into a model. This is the point where this master thesis takes place. Based on the concept of [3] where different views are generated out of log files this master thesis follow up by using the generated views and merge them into a model. A view defines in this work the personal sight of a person on a work process. For example a person becomes an order from his boss to gather some information of a deal. To gather the needed information this person must research the deal and summarise the information for his boss. When this process is designed as a model this personal sight is called a view. This view concept will be taken and the main task of this master thesis is to find a way to combine different views of the same process to one process model. Therefore, the following questions must be answered:

How can different personal sights (views) of the same process be merged?

How can structural factors be combined with organisational factors for a merging system?

In order to answer these questions a new approach is pursued. The start point won't be the model itself but the environment where the process is executed. In most known approaches the process is the main point to gather information which should be merged. Different theses take the structure of a process, the behaviour or the execution behaviour of processes to define what should be taken over in the merged model and what not. [4] [5] The approach presented in this work will be set more on the organisational information of a company like the organisation hierarchy to rank the different process views which should be merged. The real world brings different factors which restrict a group of people to be responsible for different tasks. Why shouldn't a level structure also help to limit the relevant information for a process?

To answer this question the starting scenario of this work is that the different questionnaires lead to a corresponding number of models. Each model represents the view which is the personal sight of a person on the same process. The main idea is that all views together show the real process so a model must be created with all the information of the different views. The only problem is that by taking all the information alone the process could be complicated because for each process it is possible that exceptions occur which makes the merged model confusing. To avoid this, a logic must be defined that filters exceptions but takes relevant

information into the merge. In this master thesis such a logic was developed to fit the described requirements.

A strategy must be established before developing a logic on how to get to the goal. In this case a research method was defined that described what to do for the developing of the logic and how to test it. This strategy is explained in chapter 2 as an analysis on which tool to use for the modelling. After that factors must be found which can be used for the logic. For this purpose potential factors must be defined which source is the model structure and the used tool. Factors alone aren't enough as they must satisfy criteria that are checked over an analysis. The whole procedure, results and the decision of which factors are used for the logic can be found in chapter 3. Now a logic must be found. The process of developing different mathematical formulas into an algorithm and how the developed algorithm is working can be found in chapter 4. Chapter 5 shows the testing process. This process is divided into 4 steps. First the test process model is introduced; second the department scenario is explained; third the different sights of the people (views) are explained and the last step is describing the generated merged model. These 4 steps are executed for three different test processes. To show that this solution isn't working on only one case the test processes are taken out of three different branches. After developing each test scenario a merged model is created and the results are evaluated in chapter 6. Chapter 7 holds the discussed open points of the work and concludes in chapter 8 future work. The related work can be found in chapter 9 and at the end a short summary of the master thesis can be found in chapter 10.

## **2 Research Method**

The research method to be conducted in this thesis must respect the following requirements.

1. A new approach for the merge should be generated.
2. The tests of this approach should be examined based on a real life process.
3. The generated approach should be used on more processes than only the testing process.
4. For the visualisation of the test models a tool should be used.

To meet these requirements it is essential to use a real world process. How did I get such a process? I worked in the company Raiffeisenlandesbank Niederösterreich-Wien, a bank and was employed in the controlling section. Therefore, I thought about taking a process which I know is used in the sector and use it for this master thesis. With the support of my superior I got the approval from the management to use a process which is called "Prolima control". This process is a real life process and the original one was given to me for my examination. Further information about this process will follow in the chapter 5. After getting the approval the example process has to be modelled. Different tools use different languages and have a different look of the structural elements like gates (gateway nodes). Therefore, one tool should be used for the whole master thesis.



The following steps include finding a tool for the modelling of the business processes. For simplicity a pre-selection on those tools was performed. A selection was made on the following programs:

- DIA
- Aristaflow
- BOC programs
  - AdoXX 13
  - ADONISCE2
  - Adoit
- Windows VISTA

The reason behind analysing only these tools is that all of them were taught in university and therefore the basic usage was known. Furthermore, the concrete details of the functions were already used so the analysis is going to be easier because there was less research to do. Another point is that the whole testing of the procedure will be on the modelling base. No programming and no automating. Everything generated in the further work will be made per hand and works on the modelling level.

In Table 1 the above mentioned tools are analysed on what elements and functions can be executed with the tool. The different analyse points are defined under the influence of the further work because the tool is important for the future work. When a factor is used for the ranking that can't be set in the tool the procedure would have no meaning because it could not be executed on a process. Therefore, to avoid this scenario this analysis is executed to find the tools which satisfy the most criteria hence can hold the most information in a business process diagram. The difficulty lies in the fact that not each tool is mainly for modelling but also for simulating. The best example is the tool Aristaflow which is a full-fledged Process Management System. With this a business process can be modelled but the main interest is to use the process for simulations. Such a model can be combined with a database and also code can be saved in an activity. DIA on the other side is a program only for modelling. DIA falls short in database connection and other relevant features like simulations.

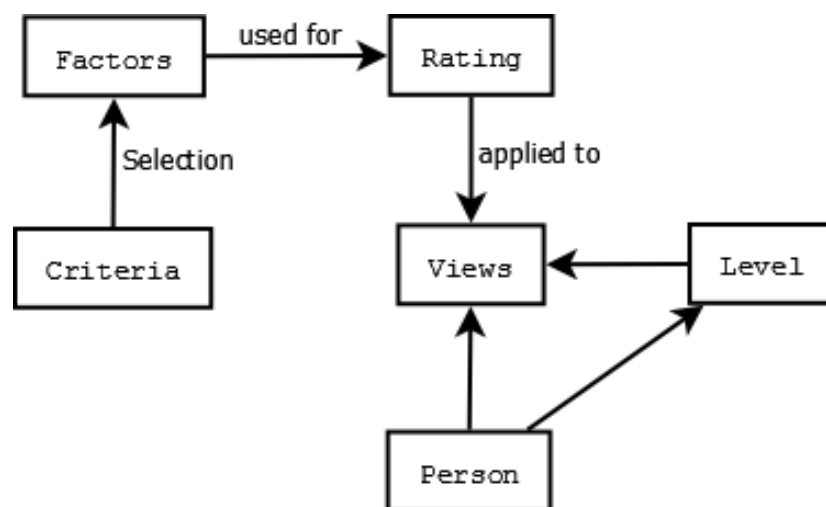
	DIA	Aristaflow	AdoXX 13	ADONISCE2	Adoit	Windows Vista
Activities	X	X	X	X	X	X
Datasets	X	X	X	X	X	X
Datafiles		X	X	X	X	
Codes		X	X	X	X	
IT-System		X	X	X	X	
Working Place			X	X	X	
Organisation Level			X	X	X	
Person		X	X	X	X	
Trial version	X	X	X	X	X	X
for free (as student)	X			X		

*Table 1: Analyse of the functionalities for each tool. The points for the analysis are defined by ideas for factors.*

To summarize the result of the analysis each tool is able to model a business process. Aristaflow and all BOC tools are capable of doing simulations. All of these four tools with simulation can combine the process with further information like codes. Aristaflow can't use information like different systems which shows in the process. The information can be saved to the activities but the only chance to see this connection is to look in the menu. On the other hand the BOC tools show such a connection in the activities. Aristaflow can communicate with different databases and also perform different codes so it is better as the BOC programs because the only code they can use is the internal program language. The advantage of the BOC programs lies in the fact that individual process models can be created with the internal language. The greatest benefit is brought by the tool ADONISCE2 in the student version because it has a package of defined models which can be used to model a whole company in every aspect of daily work, IT-systems and even organisational information. With all these information and my personal experience in addition to the fact of how long it would take to build the relevant models (because the used model type must be defined in AdoXX 13 and Adoit) for this work ADONISCE2 was chosen.

## 2.1 Conceptual Model of this master thesis

After the decision which tool to use, the literature was determined by using a core of references. The selected and further information about the used core literature can be found in the reference list. An investigation of the literature and my experience with business processes in the university and real life lead to the following model (Figure 1) which is used for this master thesis.



*Figure 1: Model form of the working process and how the used parts for this master thesis are connected.*

This model shows the connections between the single parts which are used in the further work. The purpose of displaying this by a figure is because the relationships are easier and clearer to see. The different parts are defined by the given name (for example Factors) and in the further work whenever one of these names is used it refers to this model. Furthermore, each part has a definite source, is used in a specific way and is defined to be only understood in one way. This one way understanding is important because it influences the whole work. Hence, in the following an explanation of the model for each single part takes place. At first an

example is given based on a real life situation. Second a short introduction to each part will be given which is followed by a more detailed explanation.

For the better understanding of the model a real life scenario will be given to show how the different parts connect together. Employee A works in company X and gets a request from his boss B. He should make an evaluation of specific data and write a report for the next meeting with the higher up bosses. With this case the concerned people are defined and also their hierarchical level. These two information are defined in the model as person and level. When the employee A, his boss B and a further boss in a higher position are asked how the process for the gathering of the requested data looks like each person will tell a different story. This shows that every working process is different and is influenced from the person which is asked. These personal sights are defined as views. The purpose of this master thesis is to take the defined different views and merge them into one business process model to show the complete process. Therefore, in the following work factors will be defined which are used to rank the different views. The ranking is the goal because if the views with the important information can be found the merge will have lesser outlier. For example for the explained request the higher boss which takes part in the process will only have a request and receiving activity just as well as the boss B of employee A. This means the views of these two people have lesser useful information for the working process than the view of employee A which is producing the output. The last part of the model is named as criteria. These criteria are defined rules which shall be satisfied by the factors. The criteria are there to help find factors which have the quickest and detailed effect for the ranking. After describing the model by using a real life scenario the second part of explanation will follow now:

- Criteria: A Criterion is defined and only serves the purpose to help to select the defined potential factors. A further selection analysis will take place in chapter 3. Only the factors which satisfy part of the defined criteria will be chosen. Why only one part and not all? The answer to this question will be given further in this chapter.
- Factors: A Factor is a characteristic number which holds one specific information about a part of a business process. These factors will be extracted from the literature and the used tool ADONISCE2.
- Ranking: The factors which are selected under the criteria will be used to define a logic to rate the generated business process views. The outcome of the logic should be that single business process views are selected per ranking to be used for a merge and others will not.
- Views: A process is defined by a person. So the content of the process is influenced by the person's ability to finish the process alone. Also, the sequence of the activities is influenced by the working process of the person. If the person likes to take it easy the quicker and easier activities will be executed first but if the person likes to take the heavier part first the sequence will be different. So a view declares the sight from a specific person on a process. Further information can be found in [3]
- Person: The explanation for a person is simply the holder of the design of a view. One person can have one view so each view can be assigned to a person. But the views

mustn't be unique because of the limitation from the rights for example on tools it is also possible that two people on the same level have the same view on one process.

- Level: As explained every person is different in behaviour. The process can also be different because of external effects. The first effect for a person in a company would be the position in the hierarchical structure. The position goes hand in hand with duties, responsibility and as a result the assigned rights on data and programs. So the hierarchical level is essential for the design of the process views because of the rights.

### 2.1.1 Criteria

After defining the single parts of the model a more detailed explanation of the connections between the parts will now follow. First of all the model is divided into 6 parts. These parts are generated from two sources, the literature and my practical experience. The start point in the model is criteria. The used criteria for selecting the found potential factors are defined by my personal experience from coding and modelling. Which criteria were defined?

- Identifiability
- Automatic generation
- Checkability

Why these 3 criteria? First of all "Identifiability" is the most important criterion because it is nice to have many criteria defined and make a logic for ranking and merging, but when the needed factors which were used can't be generated out of the known information the logic can't be used. So at first should be analysed which factors can be found in the process and further saved information. Using this approach avoids the trouble of finding out that a factor can't be identified and readout at a later time. Next criterion "Automatic generation" is a criterion which is only a nice-to-have because at this point of work it can't be defined that only automatically generated information exists and can be used for the ranking logic. Therefore, this will only be a nice-to-have criterion. The last point "Checkability" is needed to define that any further used factors must be checkable. It is not necessary that the factors must be automatically checkable but a human should be able to check the used factors. The reason for defining this last criterion was set under the idea that the outcome of this master thesis will be a merged business process but this model must be controlled by a human at best from a business process modeller. The controller must also be able to check the way from the different views to the merged model if the process should be comprehensible. Therefore, the factors must be controllable in a way.

### 2.1.2 Factors

The factors are defined by two different sources. On one hand the literature was analysed to find the factors which are used at the moment because to develop a new approach it must be known what the current basic approaches are. The current situation can be found in chapter 9 therefore the situation can only be summarized in one unit of information. The model holds structural information such as activities and data information. With these information similarities can be found when combined with an algorithm and or ontologies. So the structural factors are extracted from the literature. On the other hand the organisation level and System architecture information are extracted from real life which I experienced myself. Further information about the organisation structure, limitation rights and used system will be

defined by me. The next step includes an analysis of the found potential factors which of them satisfy the defined criteria. Because of the usage of the tools during the academics studies and the working in a company each factor is well known thus a personal influence can't be omitted. In conclusion, an objective analyse isn't possible therefore no effort will be put to try to make this kind of analysis. So in the end an extra point will be implemented in the analyse which is my personal experiences with the single factors.

### **2.1.3 Rating**

When the decision is made which factors will be used and which won't the next step includes the development of a logic which helps to rank the views. The ranking will be used to decide which views are going to be put into the merged model and which will not. The first idea for such a logic is a simple summary of the used factors but the evaluation in the further work will show if this approach brings a satisfying outcome or not.

### **2.1.4 View**

A view defines the personal sight from a person on a specific process. This means that a process is influenced by the person who designs it because each person has different progressing methods. These methods are essential for the design therefore each person has a different sight of the same process. This doesn't mean that each sight is unique. The design of the process also differs because of the rights which are assigned to the person. If the person doesn't have the right for a program which is used in the process then this shows in the process with an activity or more structural elements for example a sub process. So people on the same organisation structure position with the same rights can have the same view but there is also the personal effect which influences the sequence. To summarize it a view shows the personal sight of a process under the influence of organisation structure level and the included rights of this position. Further information can be found in [3].

### **2.1.5 Person**

As already explained a view includes the existence of a person which lead to the view therefore the reality that a person exists can also be used to count it in for the merge logic. The next consideration is to find out which benefit can it bring to design a logic which includes a person. To answer this it must be analysed which information can be got out of a person. First a person works for a company therefore the duties and responsibilities and also the branch in where the person works can be noted. These are interesting information because it includes a position in the organisation structure of the company and a limitation on specific information. For example a person in a bank can be in the market risk management and analyse the clients or be in the balance sheet accounting and only knows about the situation of the company but not of the individual clients. So these two people are in the same company but have different programs and therefore different rights on data. In conclusion these people never work on the same processes and can be identified due to the assigned rights. So by using the fact that a person is needed to get views it can bring benefit in form of potential factors for the merging logic.

### **2.1.6 Level**

As explained in the views and person an organisational aspect will be taken into account for the logic. Therefore, the logic should be a ranking system which includes the idea that designed views will be levelled under the condition of factors. A level shows in every aspect of

the main idea of this work. So which level is examined in this model? The level modelled in the definition model symbolises the organisation structure because of the importance of this external effect and the great influence on the design of the views it was modelled separately. The reason behind this act lies in the fact that not only the influence is great but also that the level depends on the person so a direct connection exists between a person and its level. Therefore, a direct link goes as well from the level to the views. Why is this connection so important? Simply said because it will help by the ranking but how can it help? This is easier to explain with an example. The situation for the example is the following. A company with 4 organisation levels was interviewed. The searched process takes place in level 3 and is executed from only 3 people but the interviews take place on each level. So now if the information about the level is combined with the view it's simple to tell that the levels 1, 2 and 4 are irrelevant and only the views from level 3 should be taken into consideration. So to accomplish this kind of deduction the organisation structure was worked into the model as the point level.

## 2.2 Further examples and evaluation

The development of this new approach is going to use the organised process from the company Raiffeisenlandesbank Niederösterreich-Wien. Therefore, all the tests and updates for the factors and logic will be based on this one example. The assumption that this logic will only function on this one process is self-evident. To refute this thought two other examples will be used for testing the logic. The outcome of the three examples will be evaluated and the outcome of this evaluation should be that the approach can bring a similar result to all of these test examples. Nonetheless, the evaluation can only be successful if the generated merged models can be compared with something. For the process "Prolima control" the original recorded process was organised for this purpose. The other two examples will be based on a task of a working sheet from a university course (Workflow Technologies, SS14) of the University of Vienna. Therefore, the original process will be the task of the sheet and the views will be generated based on the original by me. So the other two processes will not be real life processes but simulated ones which are shown in real life. Further information for these situations can be found in the paper chapter 5.

## 2.3 Introduction of the used tool

The used tool for this master thesis will be the ADONISCE2 from BOC [6]. This tool is used primary because of the integrated defined model types and with the help of these models a whole company can be modelled. The mentioned models are the following 13:

- Company model
- Business process diagram (BPMN 2.0)
- Choreography diagram (BPMN 2.0)
- Conversation diagram (BPMN 2.0)
- Business process model
- Document model
- IT system model
- Product model
- Working environment model
- Risk model

- Control model
- Use case diagram
- Business process diagram (BPMN)

What information is included in every model is easy to see because of the expressive naming of the model types. Now a brief overview over the connections between the model types will be given. Between the models there can be set connections so the information is not redundant and also helps to get quickly up and down in the company structure. With the company diagram the organisational structure as departments can be modelled. These company models can be linked among each other and also to business process diagrams or models. In a business process a connection can be set to data information which is created in document models. It's also possible to include the relevant IT systems which execute activities in the process. For this purpose an IT-system model must be constructed. A working environment model includes organisational positions and people to tasks in the business process. A product model holds the information over all products the company provides. Risk model and control model are security models. Use case diagram holds information about system requirements for different roles and their functionalities and also the connections between the functions. Last but not least the choreography model and conversation model will be explained. A choreography model is started with an event and includes the organisation of tasks which occur after the event. A communication shows simply a conversation between single stations or the whole company. These two models were explained at the end because from my experience I used these only once when I learned the tool.

The relevant model types for a business process are a business process diagram, or model, document model, IT system model and working environment model. With this selection each information of the process can be designed and directly linked to the business process model. The other model types are relevant for a whole display of a company but this isn't necessary for this paper therefore only the relevant ones will be used for the following development of a ranking system.

## 2.4 Summary about the research method

To summarize the research method it is to say that "Prolima control" was organised first from a real life process from the company I worked at. In the next step a model was developed which shows the connections between the different steps which are going to appear in this master theses to finish the goal of the task. Criteria were defined to help to select from a list of potential factors those factors which are used to create a logic for a ranking system. This ranking will be used to select the personal sights of people - the so called views. So a view has a person as base the person executes a process in a company. Therefore, because of the company there is a structure in which the person is integrated on a specific level so this level will also be used for the ranking. The outcome of the whole work is a merged process model which includes the information of a number of views which are selected with the ranking system. The benefit of this system should be that the views with most information should be taken and exceptions can be excluded from the merge. To show that the developed ranking system and merge are not a specific solution for only this example two other examples will be tested and an evaluation over all three examples will take place. With this evaluation it should be clear that the logic can be used in many situations and different branches.

### 3 Factors

The aim of this chapter is to identify factors for the ranking. In the first step the literature and the described tool will be analysed to find potential factors which may be able to be used. After that in a second step the identified factors will be selected under the condition of the defined criteria into which factors are useable and which are not useable. As explained in chapter 2 the defined criteria are the following three points:

- can be identified (Identifiability)
- can be automatically generated (Automatic generation)
- can be checked (Checkability)

Short summary: In this chapter the following two questions will be answered:

1. What potential factors can be identified from the literature and tool?
2. Which factors are useable for the ranking under the condition of the defined criteria?

#### 3.1 Identifying and Selection of factors

In this part the literature and the used tool ADONISCE2 will be analysed to define potential factors which could be used for the ranking. These potential factors will be evaluated to find those which satisfy the defined criteria. After the evaluation an explanation of the developed point table which is needed for the ranking system will be given.

##### 3.1.1 Analysis of the literature

The references give a good overview about actual approaches to find similarity in activities (labels of activities) and how to change these similarities to result into equality. For example [5] shows that it is possible to identify words in the label of two activities which are singular and plural and changes the plural into singular. Furthermore, by using an ontology, synonyms for the same meaning can be identified and changed to the same expression. To summarize: the most often used factors in the references are the number of activities and labels for a similarity check.

Another method for merging in the references [5] is to combine structural aspects such as the order of the activities and the connections of the data sets (meaning a data object which can hold any information like a database table, a file or a variable value), and behavioural aspects such as the content defined about the labels of the activities. [7] In more ways, synonyms and differences between singular and plural of the words were eliminated to find activities with similar names. [5] The problem of using similar names which are different but means the same for an algorithm is well known. Therefore, it was decided that without a further analysis of the possible ways to identify similarity it is not possible to decide which similarity finding method would be the best for the developed ranking system. [4] Such an analysis would consume too much time and would need a greater amount of testing examples therefore a rule was defined. This rule includes the fact that in the whole following master thesis the assumption is set that if the names (labels) of two activities are the same this means that they are identical.

##### 3.1.2 Analyse of the used tool ADONISCE2

The analysis is based on the specific used tool ADONISCE2 but most of the potential factors can be used in other tools such as the analysed tools from the chapter “research method”. The tool can model a whole company by using 13 different model types as already described. For



the purpose of this work only the models which can be linked to a business process are relevant therefore only the following models will be used:

- Business process diagram
- Document model
- IT system model
- Working environment model

Starting point for the analysis is the business process diagram because in the following work all process models and views will be modelled with this model type. First of all the structural characteristics of a business process were analysed. The following structural factors could be identified:

- Number of activities
- Number of data sets
- Number of sub processes
- Number of data connections
- Number of connectors (gates)
- Number of “or” gates
- Number of “and” gates
- Number of other gates

After developing this much information from the structure of the process there can be even more information in the details of an activity in the tool. On the surface of the details can be noted, connections to other model types to gain more information. If a link is set, the data is directly linked to the origin in the model. With this knowledge it is theoretically possible to extract this link information automatically.

Following the other two model types were analysed. The first to be analysed was the document model. Further information was extracted about the differences between internal and external files. Whether working with files the document model would be used to bring the benefit of a direct link between a specific data object in the document model to an activity in the business process model. If an IT-system model is also linked then extra information exists. For example with the link to an IT-system it could show that an activity is executed only from the system and doesn't need human resources. Also available are information like the number of activities which only need system resources. With the links set to the needed systems also the number of connections to a specific system and total number of connections to systems can be extracted. It must be registered that the number of connections to a specific system is hard to tell because the single system must be known beforehand to extract this information.

In a working environment model single work positions can be modelled. The executor of this position can be included but it is also possible to not define the person on this position but rather the working role. Working role is meant as the defined role which is used for the duties and responsibility of a position in the structure. What is not possible to model is the connections between the single working segments what brings the disadvantage that no department can be modelled but only the working places.

So with this the personal experience with organisation structures takes part. The situation of all used company examples was analysed to find potential factors which could be used for a ranking system. The analysis was performed with the help of modelling the organisation structures. Furthermore, the similarity between the different structures was analysed and defined as potential factors. The outcome of this analysis follows in this chapter.

Why was it decided to use a ranking system? The answer is as simple as can be because using a ranking system should help to avoid exceptions in the views. The hierarchical structure is defined by levels. Each level has defined rights and work. On one side the higher ups don't do the work of the lower ranks and on the other side the lower ranks only deliver their results to the upper levels but don't know the following process. So this all means the higher ups don't know the correct process and the lower ranks don't have the rights to do the total process because they only deliver part of the result to higher level. If this level system helps to assign important works which holds high responsibility to the higher ups over a step system then it should be theoretically possible to use this system for a ranking system for views.

What information can be extracted from these experiences? Following are the potential factors that could be extracted with the help of an analysis from the known organisation structures (the factors are based to be extracted from a single view):

- Number of involved people
- Number of involved departments
- Level in the organisation structure of the person and the views level (organisational pyramid level)
- Limitation level for the person and the views level (rights)
- How often does the person execute this process over a defined time interval?
- How often is this process executed over a defined time interval?

These are simple questions which every person can answer quickly but for a machine it is difficult when no data pool holds this information. The most important fact is not defined yet that the overall main rule must be that each view which should be merged must deliver the exact same output as every other view.

### **3.1.3 Evaluation of the potential factors**

In this chapter each of the found potential factors will be analysed on this defined criteria:

- Can this factor be automatically generated and checked?
- How can this factor be used to help to rank the views?
- In which combination with other factors can this factor be used to rank the views?
- What are my personal thoughts to this factor?

Please note that in the following analyse of the potential factors a big part is influenced by my personal experiences with modelling tools and working in different companies. This leads to the point where these decisions based on the analysis are strongly made out of personal opinions.

#### Number of activities:

1. Can this factor be automatically generated and checked?  
Under the assumption that the graph is saved in any form of textual logic this factor should be easy to be automatically generated with a simple program code. The nature of any automatically defined factor is that it can be also checked automatically. Therefore, both the generation and the check of the data correctness can be guaranteed.
2. How can this factor be used to help to rank the views?  
With the number of the activities it can be checked if the processes differ in their size. The size of a graph is straight edge to the number of activities. Because start and end are fixated on one per model therefore they don't influence the size of a graph.
3. In which combination with other factors can this factor be used to rank the views?  
Only in combination with a loop or other factors this criterion can be used to help to rank a view. This is because the number alone can tell the difference in number of activities but it cannot differ between the options that an activity might be deleted and another is being created. In this case the number of activities is identical but the activities differ and this important fact cannot be identified only from a number.
4. What are my personal thoughts to this factor?  
In my opinion I think this factor is not suitable for a ranking system. The number of activities indeed is a good factor for identifying the difference in size and also similarities between two views. But this is the only information that can be provided from this information it cannot differentiate if the single elements are identical or not what is too important. Therefore, this factor isn't going to be used for the ranking system.

#### Number of data sets:

1. Can this factor be automatically generated and checked?  
It is also part of the structure of the graph and if this structure is saved in any form of text than the number can be counted out of this text. Only requirement is that the different types of elements are separated or can be identified with a keyword. If the automatic generation is a success a check is a simple thing to implement.
2. How can this factor be used to help to rank the views?  
The number of data sets can help to check the difference between the views. If an extra data set is used or the number has decreased it leads to the fact that the factor alone is only a number and that it cannot tell if the available data sets are identical or not.
3. In which combination with other factors can this factor be used to rank the views?  
Only in combination with other factors, like number of links set to data sets, this factor is useable. It's the same as the factor number of activities. It can give a brief overview over the total number of existing data sets but the important fact of equality cannot be accomplished from only the number.
4. What are my personal thoughts to this factor?  
The number of data sets in my definition is a file or a variable so with only a number it cannot differ between the single elements. The number of data sets can increase or decrease because of the rights of the person. So it could be deduced that the rights of

the person differ but for the rights an extra potential factor is defined. So this information is not absolutely needed for the ranking system. Nevertheless, the data sets shouldn't be totally ignored.

#### Number of sub processes:

1. Can this factor be automatically generated and checked?

The number of sub processes only can be gathered when a keyword is saved that identifies an activity as sub process. In different tools a sub process has another design which leads to the problem that in special cases an activity is hard to tell apart from a sub process. So a special keyword or another identifying method must be present that the factor can be generated and also checked.

2. How can this factor be used to help to rank the views?

A sub process is an important information because it has a greater cycle time and can have two reasons to be used. First the sequence is often used and not only from this process so it is set external to avoid redundant information. Second the cycle needs higher rights or other rights as for example rights to read external files. Nevertheless, it is important for the graph to know how much sub processes it has. The number can't tell if the sub processes are equal but in this case it necessarily mustn't be known. This is because only the existence of a sub process is a sign that it goes hand in hand with complications as explained in the situations which lead to a sub process.

3. In which combination with other factors can this factor be used to rank the views?

Only the number of sub processes could be enough when it would be assumed that each sub process exists because the rights differ. This case is not very likely to be adopted so without a combination this could lead to an at the moment unknown problem for the ranking. The only mix with a reasonable basis may be the number of sub processes with organisation level and limitation level. The organisational structure and rights level can help to deduce if the sub process is in the view because the rights are too low. Otherwise, the deduction is that the sub process is needed because of the behaviour from the process.

4. What are my personal thoughts to this factor?

For the purpose of ranking, this factor is interesting because it can help to rank the views over the information of this number in combination to get a hint if the limitations are the main influence. Nevertheless, because of the separate defined potential factor limitation level this factor is not beneficial in using because it is more complicated than the other factor. Nonetheless, this factor shouldn't be completely ignored.

#### Number of data connections:

1. Can this factor be automatically generated and checked?

The number of data connections can be automatically generated because it must be saved in any way in the background and on this information an algorithm can work to count it out. Simpler is it by using a human who count the number personally out. The check can also be made in both ways but the reason why a machine is better is because when the process is big and has many connections it is problematic for a person to find his way through without getting lost.

2. How can this factor be used to help to rank the views?

This factor can differ between the numbers of data connections. It is not relevant to know the number of data sets to deduce the situation of the user. With situation is meant that each member of the organisation has different rights as already explained. This leads to an increase or decrease on data connections because single data sets can't be used or are additionally needed.

3. In which combination with other factors can this factor be used to rank the views?

It is a given that more rights don't mean that he has lesser data connections but in combination with the organisation level it can be deduced that the member can have higher or lower rights.

4. What are my personal thoughts to this factor?

This information is interesting because the difference in data links is not only because of the increase of data sets. The difference can be because of different rights or an additional check. So the cause cannot definitely be defined. Therefore, it is a good starting point to differentiate between the views and will be used. This factor in combination with the limitation level will help for the ranking system.

#### Number of connectors:

1. Can this factor be automatically generated and checked?

How many decisions are made in a process can also tell a part of the story which view is bigger than another because a decision includes not only the decision connector but also two paths and minimal one condition. This means that the process has more than one difference in links activities and also possible data sets and connections. To extract the information needed for the merge includes not only the connector itself but also all paths and the elements on these paths. To get so much information require that the graph is saved in some kind of text. Only with the structure saved, it is possible to extract the factor and the needed information which must be merged.

2. How can this factor be used to help to rank the views?

Connectors set a very important anomaly in the structure of the process not only does it divide the path in two it also has activities which are different on these paths. Last but not least the two paths must be reconnected to one to wander further in the process. Can it be used for a ranking? It can't be count usable because the main reason for using a connector is because of circumstances in the work and the further development of the data but not because of structural issues. Another reason might be because of too low rights but in the end it can't be told clearly why a connector is used.

3. In which combination with other factors can this factor be used to rank the views?

Even mixed with the number of activities and number of data sets this factor cannot be used to rank the views because the number doesn't only increase because of the rights being changed but also if the person is over correct and make many tests which can lead to connectors. So the reason why the number can increase or decrease is mostly based on the person who executes it.

4. What are my personal thoughts to this factor?

The number of connectors can definitely be used although it is unstable because the cause for a connector can't be cleared. If the structure differs in size than an increase in the number could help to identify such a connection. So the benefit lies in the deduction of the change of other factors.

Number of or-gates:

1. Can this factor be automatically generated and checked?

The number of or gates can be simply count by a person which looks on the graph but under the assumption that the structure is saved as text it also can be gathered automatically and also checked. Additional to tell is that there are tools which can divide between or gate types as for example an exclusive or gate. This fact was noticed but not included in this analysis because it would require an analysis which defined or gate types in the tools exist. Therefore, every or-gate variation is going to be counted as or gate.

2. How can this factor be used to help to rank the views?

A reason to use this factor for ranking couldn't be found because only the number of or gates isn't specific enough to deduce any definite reason for a difference. Furthermore, the information if the gate type is an "and" or "or" brings no benefit for the ranking system.

3. In which combination with other factors can this factor be used to rank the views?

No reasonable combination could be found because the factor alone cannot be used properly.

4. What are my personal thoughts to this factor?

I think that the number of gates alone is enough information. For the structural modification it is irrelevant if the gate is an and- or or-gate. The fact that no combination or reason for using this factor could be found backed this opinion up.

Number of and-gates:

1. Can this factor be automatically generated and checked?

If the structure is saved in any form of text then the information can be automatically generated with a simple algorithm. Also, a check can be performed with the same algorithm. Nevertheless, it is also very easy for a human to count out the relevant gates so why not using a person. Because a machine works also on simple graphs as on very difficult and complex processes the same way and approximately the same time. But the bigger processes are a problem for a human due to loss of clearness.

2. How can this factor be used to help to rank the views?

Only based on the information of and-gates no meaningful reason could be found that this factor can help for the ranking.

3. In which combination with other factors can this factor be used to rank the views?

No good and meaningful combination for a ranking could be found. The information is not significant enough.

4. What are my personal thoughts to this factor?

This factor is the same as the or-gate. The factor alone is not of any value for the merge because it isn't of any use to know if the gate is an and. This only is interesting for the behaviour of the process.

#### Number of other gates:

1. Can this factor be automatically generated and checked?

Every information that can be seen in the structure can be automatically generated under the assumption that it is saved in any form of text and explicit keywords are used. Therefore, this factor can be automatically generated and checked under the condition that the keywords are distinct and known.

2. How can this factor be used to help to rank the views?

This is a particular factor based on the thought that each tool can perform different forms of gates. The best example is the modelling language BPMN which can perform many different gates. [8] Nevertheless, a definite reason for any use of this factor couldn't be found.

3. In which combination with other factors can this factor be used to rank the views?

The factor in single couldn't be used and no meaningful combination was found.

4. What are my personal thoughts to this factor?

This factor was only included under the aspect that different modelling languages can have more gates or other gates. But for the following examples this factor isn't needed because no such gate is used therefore the factor will not be used.

#### Number of internal files:

1. Can this factor be automatically generated and checked?

Under the condition that it is possible to define in a document model if the used data sets are internal or external it should be possible to get the number of it automatically out of the process. This conclusion is based on the thought that by using a document model a data set exists which is connected to the process model per link. So in the business model must be the information that the link exists, without it it's not possible to generate this factor. When the links can automatically be extracted it's also possible for a check. But when the link to a document model can't be set than a definite way to model a data set as external is from my experience not possible. Therefore, an automatically generation or check is not performable.

2. How can this factor be used to help to rank the views?

It is an important information if the used files are internal in the organisation or must be ordered from external. The reason for this is that mostly the used data must be on the latest update for the process so the executing person has to request the latest version every time when the file is external. The problem is that the number only gives the information that in this process there are internal files in use but it cannot identify if there are other files or even what the total number of used data sets is. Therefore, this information alone cannot help for the ranking because it's not clear why the number can differ between the views.

3. In which combination with other factors can this factor be used to rank the views?  
First of all the total number of data sets must be known to use this factor because without the total number it can't be measured if there is a difference between two views which can exist due to a change in the number of files. The next step is to identify with the factor number of external files and internal files how much external, internal and other data sets exist. This can only be achieved if all 3 factors are defined and combined and could be used for the ranking.
4. What are my personal thoughts to this factor?  
Most processes that I know only have internal files because the data which is used from more than one department is located on an extra device where either departments or more have access. This means for me it is after all an internal file but more than one group use it. For me this is not a factor which I would use because a file is a file after all independent of external or internal ordered. The found approach that more external files leads to a higher risk will not be followed because this would require an analysis to proof if this thought isn't only theoretically.

#### Number of external files:

1. Can this factor be automatically generated and checked?  
Under the same already described aspects by the Number internal files this factor can be automatically generated or checked.
2. How can this factor be used to help to rank the views?  
This factor is important for the ranking because it represents a difficult risk for the process. In order to use an external file special rights are required and it must be controlled if the used file is on the last updated state or not. This is because when the file isn't on the latest version it could provide a false output of the process. So the risk of using external files is given and must be shown in the modelled process. If there are external files used it is important for the ranking.
3. In which combination with other factors can this factor be used to rank the views?  
As already explained by the factor of internal files the best combination is to take the number of external, internal files and mix it with the total number of data sets.
4. What are my personal thoughts to this factor?  
A problem I identified is, that it must be defined what an external file is. Is it a file from another company or from another department? The difference is that from another department the actuality of the file can be got easily. But from another company the file will mostly be ordered over an interface. That is very hard to communicate to the other party that the file must be updated or if the data in the file is not correct. Moreover, if it is wanted to use an external file the person needs access and normally a step in the process which shows the access and the return value. So the only point assumed for using this factor is the knowledge that one activity will occur and the person must have a different right to others. The fact that another defined factor is the limitation level prevented me from using this factor because the rights alone can tell clearer than the assumption which could be made with the help of this factor.



#### Number of system activities:

1. Can this factor be automatically generated and checked?  
When the structure of the process is saved in a form where an algorithm can readout single information and this factor is filled in the model than this factor can automatically be generated and also checked.
2. How can this factor be used to help to rank the views?  
The number of activities which are only realised from the system are a good information to take for a risk factor. Furthermore, if the number of activities with human resources is high the duration time can differ very strong and from my experiences the possibility of mistakes is also growing. This means that if the process has high number of system activities the difference between the views can be lesser. Nonetheless, only the information of system activities cannot provide the similarity of the activities between the different views because the number may be not different but single elements can differ.
3. In which combination with other factors can this factor be used to rank the views?  
No real combination for a ranking could be found because the information alone is good to explain big differences in duration time or other time differences but not to rank the views because the significance of a view is not measured over the duration time. Furthermore, there is no incontrovertible argument to rank a view with more system activities higher than others.
4. What are my personal thoughts to this factor?  
Personally I think that system activities are good for merging, because the possibility that the definition of the activities is the same is higher. That concludes into the theory that more system activities makes the similarity check easier but this doesn't change the fact that both activities will be treated the same. For the merge only similarity is essential and not if it has human resources or not, that goes under the behaviour. Therefore, the difference in behaviour for example duration time is not needed for the merge itself. For the ranking it can be taken as a factor but then only in combination with the total number of activities for comparison.

#### Number of links to the systems:

1. Can this factor be automatically generated and checked?  
A link to a system is not graphically visualized therefore it is relevant if the information even is saved or not. The bigger part of tools I used so far didn't take in account the difference between the systems or show it. The only tool which combines a process and a system is the used tool in this work which allows linking an activity to a system in an IT-model. Theoretically this information could be readout but to check the correctness of this factor is for a machine impossible. It can be checked if the link leads to an existing system and if the system is online but not more. If it is the correct system or not can only be told by a human because he can deduce this information.
2. How can this factor be used to help to rank the views?  
The definition of a link to a system in this work defines an access to a system over an interface. This information can be a hint that external data or a database is being needed, etcetera. But in which case this information can be used for ranking could not

be found because the number of accesses to IT systems only shows that a right is needed but not if it is special or not.

3. In which combination with other factors can this factor be used to rank the views?

So in combination with the limitation level it shows if special rights are needed for this system. Nevertheless, the limitation level alone can also show this and furthermore alone not in combination.

4. What are my personal thoughts to this factor?

It is difficult to tell the links to systems only on a visual part because a link in the graph symbolizes only the need of a data set but not if this data comes from a system, an interface or is located local. This leads to no concrete information therefore this factor is hard to get and in cases impossible to get. To summarize it up: the factor will not be used.

#### Number of involved people:

1. Can this factor be automatically generated and checked?

Under the thought that when I used tools it was possible to define a person in charge it can easily be counted how many people are involved in one view. The only condition is that the structure of the view plus the extra information is saved in any form of text and a keyword exists that identifies the person in charge. When the information is gathered automatically it also can be checked automatically but only if the extracted information is correct.

2. How can this factor be used to help to rank the views?

The number of involved people in one view is a very interesting factor because it can be assumed that the reason behind a higher number than one is that the original person doesn't have the appropriate rights for this process. With this assumption we can deduce that we can rank a view with more people lower as a view with fewer people.

3. In which combination with other factors can this factor be used to rank the views?

This factor alone is already specific enough so that a combination isn't necessary. Anyway if the information of the organisation level from the other person would be used it can give an interesting extra aspect. The view could now tell whose permission is needed to develop further in the process. This information could help to identify the people who are indispensable for the further merge.

4. What are my personal thoughts to this factor?

This factor can definitely be used for a ranking system because of the statements in point 2 and 3 this factor can help to rank the views easy. Furthermore, if the other involved person would get more attention, then it could help to clear up the gathered outcome of this factor. If this other person has a view it would be ranked very high because it assumes that this person is indispensable for the process. Nevertheless, if such a case would carry on to control if the involved person has a view this would lead to the fact that a process with more people would get complicated. With this thought in mind it leads to the fact that when 10 people are involved also 10 variables must be declared which save the information if there is a view for a specific person or not. That would be difficult to count and imply when making the ranking concept. Nonetheless, this factor should be used.

#### Number of involved departments:

1. Can this factor be automatically generated and checked?

It is difficult to define if it is possible to gather this information or not because even if the information of the involved workplace is needed it doesn't mean that the department is also saved. Therefore, it might be possible to combine present information as the workplace information, organisation level and limitation level to deduce in any way what departments are involved. The great problem lies in the fact that all needed information for the generation are hard to get and mostly better off to be taken care off from a human resource. That includes the increase of possible failure but would be the quicker way. A check for this information might be likewise as hard to make as the gathering of the information itself therefore a human would also be the better choice.

2. How can this factor be used to help to rank the views?

The number of involved departments can be useful because this means that the working process is based on data which is needed in more than one department. That concludes that more than one person needs the output of this data and there might be a high possibility that in one or another way a second person is needed for this process to be completed. This is only an assumption so it can't be proved what leads to the conclusions. Therefore, this factor can't be used alone for a ranking without prove.

3. In which combination with other factors can this factor be used to rank the views?

If this is mixed with the information of involved people it could lead to the information that a different department and an external person might be needed but this definitely can't be deduced only based on these two information. So the number of external files should also be used. It also can be that a file from a different department is used and the extra person is not from another department. So this factor can't be used in singular or in combination for the ranking because it isn't unique and only good in combination with many other factors.

4. What are my personal thoughts to this factor?

Since the number of involved people is already used the number of departments is not needed. The increase of involved people alone is a good factor. It is not important if the people are from different groups or of the same. It shows simple that the person alone has not enough rights or permission to work alone on this process which means that it should show in the ranking. This factor is in my opinion not needed when using the number of involved people.

#### Organisation level:

1. Can this factor be automatically generated and checked?

The only tool I ever used to model a working environment (ADONISCE2) don't show a fully hierarchy it only shows which work places are set and who works on these places. Therefore, this information is nowhere saved and can't be generated automatically. The check is even harder to make when a human makes the plan because the correctness of this model is hard to keep due to the continuous changes in the human resources. The level of the organisation shouldn't change so much therefore this information is easier to update but the check can't be guaranteed.

2. How can this factor be used to help to rank the views?

The organisation level shows where the views' person stands in the organisation structure and can help to understand the workload of this person. With the help of this information the organisation level alone can rank very easily the different views.

3. In which combination with other factors can this factor be used to rank the views?

The combination of organisation level and limitation level could quickly provide a ranking under the condition, that it is possible to restrain on an area. That's because the easiest way to rank the different views is a defined sector where the process takes place. Under the assumption that there is more than one view in the same organisation level other factors must provide the further ranking.

4. What are my personal thoughts to this factor?

The only problem is that for a limitation of an area the total number of levels must be known. So if this number is defined the level which is important for this process can quickly be defined. For the purpose of ranking, this factor holds an important information because if it is clear which level is the needed one all other levels can be ignored or lower ranked for the merging. Due to the difficulty to get this information the best method is to let a human define the levels and keep the correctness. With this system a check can only be done from a human but it is possible to get the information because the organisation level should not change very often.

#### Limitation level:

1. Can this factor be automatically generated and checked?

The limitation level shows the level of rights a person has but the only method I know to get more rights for a person is to add a right out of a list to a person. This means that the rights are saved in form of a list and not on levels but if there is no hierarchy in the rights where they are saved it is not possible to read out the levels. If it is combined with the workplace it may be possible to combine the information into levels but also difficult. For example the used tool in this work can define a workplace model but the workplaces are in line and not hierarchical. This leads to the difficulty that a workplace plus rights must be linked to a person and with the information of employed type it may be possible to define levels but this is complicated and prone for failures. So it can be automatically generated but the generation and check are rather difficult.

2. How can this factor be used to help to rank the views?

In every organisation is a structure of responsibilities. This leads to the different rights on data's and systems. So each level has other limitations and in the different levels more than one limitation can exist. So this information alone can help to interpret if the rights might be in the right spectrum of the structure or too high or low.

3. In which combination with other factors can this factor be used to rank the views?

The best combination is with the organisation level. These two factors can differ between the views with too low and too high levels to bring out the real process. In combination with the number of executions in proportion to number of total executions the views could easily be selected on the views with high possibility to represent the real process.

4. What are my personal thoughts to this factor?

First of all it must be differed between two sorts of limitations. On the one hand there are department rights which must be given to every person of the department as basis for the work and on the other hand special rights given to only a view people independent of their membership. I think this factor is essential for a ranking system because the limitation level can restrain the amount of views to an area where the rights have the needed requirements for the process. This also includes the circumstance of the two different kinds of rights because of the difficulty to generate this information it is easier to get this information by using human resources. The rights of the departments are a good starting point for the level structure because the organisation and limitation structure melt together. The individual rights should be only a few number of supplement entries in the before developed system. This is quicker made from a human and also the checking of this hierarchy is easier for a human so this factor should be fully generated by human resources and used for the ranking.

Number of executions:

1. Can this factor be automatically generated and checked?

It's not possible to automatically read out this information because this information is not stored in the modelling tool but rather in a simulation tool. The tool that is used in this work happens to can do both functions but if the information is not implemented it also isn't stored anywhere. The only assumption could be made is that when the information is implemented in the tool then it must be noted in any way together with the modelling information. In this case it is possible to automatically generate this information but the problem for this case is to get the information to note it in the tool. The only possible way found to get this information is a survey under the concerned people of the process.

2. How can this factor be used to help to rank the views?

This factor alone is rather useless because the number of executions for a period can tell nothing without a value who gives a possibility to match it with.

3. In which combination with other factors can this factor be used to rank the views?

First of all the number of total executions is essential to use this factor because now in combination the information if the person is responsible for this process or only a temporary worker in special times can be determined. Otherwise, only the number of executions holds no further space for deductions. The mixed result could help to rank the views and if it would be combined with the organisation level the ranking could be extremely detailed. This is because the proportion of total executions to executions of a person shows which views are executed often and which not. This can be displayed to the ranking. Views which are executed often are ranked higher and reverse. The outcome combined with the level leads to a further selection over the high ranked views and leads to a small amount of views with high ranks.

4. What are my personal thoughts to this factor?

This factor could help to find an improved ranking but only in combination with the number of total executions. Furthermore, if it would be combined with the number of involved people in the view and the organisation level of the views' person the ranking could restrain on a very small area of relevant views for the merging. Therefore, it

should be used. The reason why it isn't used is that for the used test processes this information isn't available and even if they are real life processes without a further investigation this factor can't be generated that easy. It would take a further investigation in the companies that are using the process and a lot of interaction with the executing people.

#### Number of total executions:

1. Can this factor be automatically generated and checked?

The result of analysing the total number of executions if it can be automatically generated and checked is complex. It is the same as the number of executions. The only possible reason could be found to get this information is per survey all involved people. Nevertheless, one different point in comparison to the number of executions is that the result of the survey must be the same under all people. If there are differences it would need a further investigation of the process to decide which number should be used. A possible way would be to take the number which occur the most. In conclusion this factor can't be automatically generated and the check is rather difficult to form because there could be different numbers out of the survey. Under the assumption that the decision which number will be taken is made from a person the factor of human individuality can't be measured on how it will influence the check.

2. How can this factor be used to help to rank the views?

For a simulation the number of total executions must be known because otherwise the simulation can't be realised. So this information is very important for a ranking system because with the help of this factor can be differed between often and lesser used processes. So for the ranking system it also can be used but not alone because with just the information of total executions it's not possible to distinguish the views because no reference value exists.

3. In which combination with other factors can this factor be used to rank the views?

The total number of executions in combination with the number of executions of the view could definitely help to rank the system. That's under the assumption that in this case the rights are not that important because if the number of executions is defined for each view it can show better than the rights or another organisation factor which view is important and which not. Therefore, if the rights are high and the execution number in contrary to the number of total execution is zero or simply very low than this concludes that the relevant view is unimportant because the rights are fitting but no execution means that the level is too high. Whereas a high number shows that this view is important because it's often in use and the person is on the relevant level for the process.

4. What are my personal thoughts to this factor?

This is a good factor for using for the ranking. The only problem lies in the gathering of this information. It must be requested from each person. No other way to gather this information could be found but this factor alone is not enough it's only of interest when combined with the individual number of executions of each person. With these two mixed information the ranking could be a lot easier because the area of responsibility would be clearer. In spite of that good usage the executions will not be used for this work because it would have required a survey in every branch of the used

test processes. This would have consumed too much time and assistance from companies.

### 3.1.4 Summary of the factors

In the outcome of the research of the potential factors one point is obvious the structural factors all can be automatically generated under the condition that the structure is saved as any kind of text. Then an algorithm can extract the individual numbers for the factors but when the criteria can be generated automatically a check is also possible. On the other hand the organisation structure can be generated automatically only with high effort because it is difficult to consider every single small exception. Therefore, it is better to let human resources declare the structure and update it. Other factors as the execution number are factors which only can be got out of a survey or some method alike a survey. The only people who know this kind of information are those who execute the process so the survey must take place on this group. Also, it must be considered that in the analysis the fact that the used tool can have different types of models which can be linked to the process and my personal experience influenced the result.

### 3.2 Summary of the Analyse

In the following table (Table 2) the defined criteria can be seen and if the individual factors will be used for the ranking system or not. Additionally, a small statement takes place for the not used factors why it was decided not to use it. In Table 2 the symbol “#” is a place marker for the text “Number of”.

Factor	Used	Not Used	statement
# activities		X	As found in the literature most approaches for identifying similarity and merging use the number of activities. To differentiate from these approaches it was decided to not use this factor and to set a higher value on the other factors.
# data sets		X	The number of data connections is going to be used therefore the explicit number of data sets is not needed because the principle is one connection for each data set so a number could be created. It's true that one data set can have more than one connection but this is only possible to count in when the number of data sets is used. This means that the result of supposed data sets can differ from the actual situation in the process but if all views are under same concept the basis of comparison is the same. Note: The other way around is worse to create because more than one data connection can lead to the same data set. Only from the data sets the number of connections can't be told.
# sub processes	X		
# data connections	X		
# connectors	X		
# or-gates		X	Number of connectors is going to be used so the type of gate is not needed to know. Each gate will be handled as simple as a structural element gate.

# and-gates		X	Same reason as number of or gates.
# other gates		X	Personally I never used any other kind of gate than or- or and-gates. Therefore, this factor is noted as existing but because of no experience and examples this factor is not going to be used.
# internal files		X	The limitation level will be used therefore the only reason to use this factor would be to decide on the rights and deduce the involved people. Nonetheless, the involved people are not explicit therefore this factor is not going to be used.
# external files		X	Same reason as for number of internal files.
# system activities		X	The benefit of using this factor would be only to know that an activity is executed by a system. But to rank system activities higher than activities with human resources shouldn't be implemented. Human resources are a point of failure but also of control.
# link systems		X	For a merge only the information that a system is used is enough. The knowledge of how many systems links and how many links leads to one specific system isn't needed and don't bring an additional value. The number of used IT systems isn't relevant under the assumption that same activities need the same system.
# involved people	X		
# involved departments		X	The number of involved people is better for using than the number of departments because it also can be that more than one person from the same department is working on one process view. In these cases the department number would provide a false effect on the outcome of the merge logic.
Organisation level	X		
Limitation level	X		
# executions		X	The main reason for not using this factor is that this factor couldn't be gathered. Without a survey this factor isn't ascertainable. The fact for using 3 examples in the following for testing means to gather for all 3 branches the information how often this process is being executed by a person. This would include a high amount of time and the support of companies.
# total executions		X	For the case study one process this factor is known because of personal experience but for the other case study processes without a survey in appropriate companies this kind of information couldn't be got. Therefore, the information isn't going to be used because the ranking system wouldn't be testable.

*Table 2: This table shows all identified potential factors and marks them with an X if they are going to be used in the ranking system. If it isn't taken, in the last row a short explanation can be found why this decision was made. This table is a summary out of the analysis which took place in this chapter to select the potential factors. The # is a place marker for "Number of".*



The views can be expanded to the absolute maximum for the table. In this work the test cases will never appear with more than 6 views which will be seen in such a table. Most important is that for the ranking two rules were declared after the analysis of potential factors.

Defined rules:

- The same label of an activity means that it is the very same activity!
- Each process view provides the exact same output!

These rules are based on identified factors which are not used for the ranking system because they cannot be generated without the support of an IT-solution. Nevertheless, these are important factors which also should take part in the ranking but are not used because it can't be controlled if the outcome of each process view is the same. These two rules are assumptions made on the fact, that without them the ranking would be harder because this information has to be taken into account. Nonetheless, it is not possible to use it in the logic therefore, it was declared as rules which every test in this work fulfils.

### 3.3 Point System

After identifying the factors and finalising the decision of what to use and what not the next step is to take the factors and form a point table. Normally this could be obtained by programming an example but in this case more than one test example will take place in the following. Therefore, programming would have consumed too much time to satisfy each case of the used test processes so the factors simply were counted out of the process views and completed into such a table, see Table 3. The real point tables can be seen in chapter 5

	# sub processes	Organisation level	# data connections	# connectors	# involved people	Limitation level
<b>View 1</b>	X	X	X	X	X	X
<b>View ...</b>	X	X	X	X	X	X

*Table 3: This table shows the structure of the future used point tables by setting the views as lines and the selected factors as rows. The X shows where the generated factor values should be inserted. The # is a place holder for "Number of".*

## 4 Ranking system and Merge

After developing the factors and deciding which to use for the ranking a logic must be developed on how to combine these factors to get an output which can be used to decide which views are usable for the merge and which are not according to the ranking. The first idea is to develop a mathematic formula which delivers a number as output which can be classified by means of a grading number such as the ones used in the school system. After a formula was defined and tested the result wasn't satisfying enough, therefore a probability had to be implemented. This probability takes place in two steps. First the outcome from the formula

was taken and multiplied with a random number between 0 and 1. Secondly, the generated random number was multiplied with single factors and not the whole outcome. After developing the formula a dead end was reached and in the following an algorithm was developed to be used instead. The individual steps of the development will be explained in this chapter.

## 4.1 Mathematic formula

The factors which were selected after an analysis in chapter 3 are:

- Organisation level
- Limitation level
- Number of data connections
- Number of involved people
- Number of connectors
- Number of sub processes

So how can it be possible to take the selected factors and put them into a formula whose outcome tells, changing the structure of the view to the merge model or not? There were a few steps which led to an end formula for this requirement.

The first idea was to summarise the extracted information numbers and decide based on the height of the sum if it's a good idea to take the change or not.

OL ... Organisation level

LL ... Limitation level

DC ... Number of data connections

NP ... Number of involved people

NC ... Number of connectors

SP ... Number of sub processes

OL, LL, DC, NP, NC, SP  $\in \mathbb{N}$

$$Take = OL + LL + DC + NP + NC + SP$$

That leads to a merge but it has one big problem. The result can differ. What is the threshold that marks the point of taking the change and ignoring everything below it? How can this threshold be decided? At first it was tested by using the average of all calculated sums. After that, different limits such as two thirds, one third, half of the greatest sum, one and a half times that of the smallest sum and one fourth were tried. Every method lead into a merge but the results weren't satisfying enough because not all important tasks were taken or too much was deleted. Therefore, it was searched for another starting point to develop a threshold.

The next step included the definition of a way to decide the threshold for using a view or not. The first methods included a definite number for the threshold so taking this into account a suitable number for this purpose must be found. How can a number be decided which can be used for all possible processes? Whatever the process branch is and how big and complex the process might be the number should be chosen based on a logic which can be used for all these environment conventions.

What if there is one base view and when for the merging only the changes are going to be inserted? The sum of this base process could be taken to determine the threshold. The only question is how to decide which one of the views should be the base one. Due to the usage of the organisational factors such as organisation level and limitation level plus the fact that a business process is limited to a part of the company it's possible to define the base. The needed factors are the organisation base and the knowledge which part of the organisation structure is responsible. If these two points are combined then the head of a group or department can be found. So the base should be the person with the highest level in the organisational part which is relevant for the process because this person has the highest rights. By taking this into account the threshold is now set with the sum of this base person.

BV ... Sum from base view

Vn ... Sum from view n

$$Vn > BV \left( \begin{array}{l} \text{changes implemented into base process view} \\ \text{view not relevant} \end{array} \right)$$

After completing this step the logic was tested. The outcome was also not satisfying enough because the problem lies in the fact that lower rights result in a higher number of structural elements for the compensation of the lost rights. In some cases an important information was in views with a lesser number as the threshold and this information wasn't taken for the merge. The reason for a lower number is that a level of rights which is too low leads to the fact that one person is incapable of completing the process alone. So the system of using this threshold which decides the usage is not sufficient and another threshold logic must be found for ranking and merging.

After looking over the used factors again and implying the fact of how to get the threshold it was clear that by using a base it also has to have definite values for the factors as base. For the base view the person with the highest organisation level is consulted. This also includes the highest limitation level. This leads to the point where it can be assumed that the other views can only have a lower level or the same at best. So these factors lead to a difference in the sum and can influence the outcome to be smaller than the base. To overcome this gap the formula was adapted to not use the whole number of the factors but the difference calculated out of the base number minus the number of the actual view. Used on the two factors "organisation level" and "limitation level" the outcome is now not so far apart. These are not the only factors so in a following adaptation the formula was modified to not use just the number but the difference to the base view for each factor.

DOL ... Difference Organisation level between base and actual view

DLL ... Difference Limitation level between base and actual view

DDC ... Difference Number of data connections between base and actual view

DNP ... Difference Number of involved people between base and actual view

DNC ... Difference Number of connectors between base and actual view

DSP ... Difference Number of sub processes between base and actual view

DOL, DLL, DDC, DNP, DNC, DSP  $\in \mathbb{N}$

$$Take = DOL + DLL + DDC + DNP + DNC + DSP$$

The problem that the defined threshold is based on the base view and the influence of the base view persists after all. No new method for a threshold could be found. All developed threshold logics were tested with the new formula. The best and stable results were achieved with the following threshold logic:

$$LV \dots (\sum_{n=0} Taken) * \frac{2}{3}$$

Vn ... Sum from view n

$$Vn \geq LV \left( \begin{array}{c} \text{changes implemented into base process view} \\ \text{view not relevant} \end{array} \right)$$

Now the sum is a smaller number but it can be used the same way as the outcome of the previous formula. However, a new problem arose with this new approach. Now the difference between the number of data connections and number of connectors influenced the outcome more than the other factors because the difference in organisation and limitation level can't be as high as the number of data connections.

To discuss the overwhelming number of connectors and connections a trial period began for using different variables instead of the sum. To make a greater effect of the organisational structure factors than the structural factors these numbers weren't summarized but multiplied with the other factors to get another effect. By multiplying some factors in this case the difference of the organisation level and the difference between the sub processes the wanted outcome should be even influenced over all factors.

$$Take = DOL * DSP * DDC + DNP + DLL + DNC$$

Why these two out of all factors? Why not the limitation level or any other factor? As already explained there is a strong connection between the organisation level and limitation level therefore using one for multiplying should be enough. The reason for using the number of sub processes is that a difference in this number might come from the difference in the rights. The base view includes the maximum on rights for this process in the case that the rights are lower than possible to extract one part of the process into a sub process. If this shows in the difference of the two numbers then it's a good factor for ranking and should have a greater impact on the outcome than other factors. After testing this method the outcome was a little too far apart to use the threshold logic anymore because the influence of the base process was too high. While using the difference numbers and multiplying a problem has appeared that this method can create a negative result. If the outcome is negative it is under the threshold and such cases will never be used. Therefore, this method would be not usable because if the negative results appear on the relevant views the whole merged process will be false because important steps are missing. So another method must be found to avoid a negative result.

How can negative numbers be avoided? The simplest method is to use the absolute value from the calculated difference numbers. This has the benefit that the number will not be changed from any side effect and the way to calculate an absolute value is mathematically defined so no difference in the calculation ever will appear. The adapted formula looks like this:

ADOL ... Absolute Difference Organisation level between base and actual view

ADLL ... Absolute Difference Limitation level between base and actual view

ADDC ... Absolute Difference Number of data connections between base and actual view  
 ADNP ... Absolute Difference Number of involved people between base and actual view  
 ADNC ... Absolute Difference Number of connectors between base and actual view  
 ADSP ... Absolute Difference Number of sub processes between base and actual view  
 ADOL, ADLL, ADDC, ADNP, ADNC, ADSP  $\in \mathbb{N}$

$$Take = ADOL * ADSP * ADDC + ADNP + ADLL + ADNC$$

During the testing an important problem was identified. By using the multiplication the outcome of the logic for specific test cases was zero. The main test object in these tests was the organisation level. The test scenario was that the two views had the same organisation level. While using the difference between these two views for the same factor lead to a zero for the formula. So the calculated outcome was zero. To avoid this scenario a simple adaptation was made. The two factors that are included by multiplying are evaluated on the fact if it is less than one. A maximum function is used for the calculation. With this method it is simple to use a neutral number which will be in this logic the number one. The maximum function takes the calculated value of the difference when it is greater than zero and if this is not true it will be the number 1. With adapting this to both factors that are included over the multiplication this problem could be solved. Now the formula looks like this:

MDOL ... MAX { ADOL , 1 }  
 MDSP ... MAX { ADSP , 1 }  
 MDOL, MDSP  $\in \mathbb{N}$

$$Take = MDOL * MDSP * ADDC + ADNP + ADLL + ADNC$$

#### 4.1.1 Result: mathematic formula

The latest version of the developed mathematical formula works with the difference between a base view which must be defined and the view which should be merged. Therefore, two specific factors - the organisation level and the number of sub processes - will be multiplied with the sum of the other selected factors. To avoid an invalid result because of the multiplication the two named factors are set into a maximum function which takes the absolute factor and use them when it is greater than zero or when it is zero then use the neutral number 1. The formula looks like the following:

ADOL ... Absolute Difference Organisation level between base and actual view  
 MDOL ... MAX { ADOL , 1 }  
 ADLL ... Absolute Difference Limitation level between base and actual view  
 ADDC ... Absolute Difference Number of data connections between base and actual view  
 ADNP ... Absolute Difference Number of involved people between base and actual view  
 ADNC ... Absolute Difference Number of connectors between base and actual view  
 ADSP ... Absolute Difference Number of sub processes between base and actual view  
 MDSP ... MAX { ADSP , 1 }  
 ADOL, MDOL, ADLL, ADDC, ADNP, ADNC, ADSP, MDSP  $\in \mathbb{N}$

$$Take = MDOL * MDSP * ADDC + ADNP + ADLL + ADNC$$

Which logic was used to decide which views should be implemented into the merge and which not? A threshold was found which defines that everything above this threshold will be taken into the merge and everything below the threshold will not be used. The comparison value is the calculated Take from the currently analysed view. The threshold logic looks like following.

LV ...  $(\sum_{n=0} Taken) * \frac{2}{3}$   
 Vn ... Sum from view n

$$Vn \geq LV \left( \begin{array}{c} \text{changes implemented into base process view} \\ \text{view not relevant} \end{array} \right)$$

After testing the different developed methods each merge which was completed had the same effect that the lower ranked rights lead to the fact that the view has a worse chance to be used than higher ranks. The problem is that also in the lower ranked views an activity can exist which is important for the merge and will not show up because the rank is too low to use it. How can this result be changed? The developed way to alter the impact of the result in such a way where the original result isn't as strong in the influence on the process as before is to use a probability. A probability which is generated randomly could help to differentiate the results in a steady way.

## 4.2 Mathematic formula with random factor

Now a random factor was added into the formula so that the merge isn't dominated from the basic view. Therefore, two different types of implementing the random factor into the formula were developed. First the result of the formula will be multiplied by a random number. Second the random number will not be multiplied with the whole outcome but only with individual factors of the formula. Furthermore, another adaptation was made to look for a better merge logic. The factor Number of connectors was substituted with a new factor named "number of check words". This factor is extracted from the labels of the activities. For this new factor there was only one check word used, the word "check" itself. The reason behind only using this word lies in the fact that the used processes for the testing were already known during this process of developing this logic. Therefore, no other check words are in the processes so no more were used for this factor. To differ between these two different approaches the tests will be identified by using the terms' formula 1 (connectors) and formula 2 (check words). Note that the source point table for the following shown test is the point table of the first test process Prolima control and can be found in chapter 5.

### 4.2.1 Result multiplied with random number

Now the only thing that has been changed in the developed formula is the integration of a random number, named RN. In the German version of excel where the testing took place the following statement generates a random number.

$$RN = RUNDEN(ZUFALLSZAHL( ) * 1; 2)$$

With adding this information the formula looks like following:

$$Formula1 = (MDOL * MDSP * ADDC + ADNP + ADLL + ADNC) * RN$$

$$Formula2 = (MDOL*MDSP*ADDC + ADNP + ADLL + ADNC)*RN$$

After testing this a while the outcome was very disappointing because the results were more random than expected. By using the result from ten tests the merges were completed and also sketched. The results of the test for the two different approaches can be seen in the following two tables (Table 4 and Table 5). The random numbers for each view where the same for both formulas in every test so it can show the logic which would bring the better results.

Formula 1		Line	View 1	View 2	View 3	View 4	View 5
test 1	sum	1,82	0,13	0	2,73	2,04	1,62
	RN		0,13	0	0,91	0,34	0,09
	decision		YES	YES	NO	NO	YES
test 2	sum	2,76	0,69	1,82	1,83	0,48	4,14
	RN		0,69	0,91	0,61	0,14	0,23
	decision		YES	YES	YES	YES	NO
test 3	sum	3,48	0,65	1,7	1,62	0,84	5,22
	RN		0,65	0,85	0,54	0,14	0,29
	decision		YES	YES	YES	YES	NO
test 4	sum	6,12	0,89	0,32	1,23	1,14	9,18
	RN		0,89	0,16	0,41	0,19	0,51
	decision		YES	YES	YES	YES	NO
test 5	sum	2,16	0,12	1,7	1,95	3,12	3,24
	RN		0,12	0,85	0,65	0,52	0,18
	decision		YES	YES	YES	NO	NO
test 6	sum	1,72	0,21	0,88	2,16	2,58	1,8
	RN		0,21	0,44	0,72	0,43	0,1
	decision		YES	YES	NO	NO	NO
test 7	sum	3,68	0,68	0,14	1,89	5,52	1,44
	RN		0,68	0,07	0,63	0,92	0,08
	decision		YES	YES	YES	NO	YES
test 8	sum	1,88	0,67	0,68	2,16	2,82	0,36
	RN		0,67	0,34	0,87	0,47	0,02
	decision		YES	YES	NO	NO	YES
test 9	sum	1,9	0,04	0,28	2,85	0,54	1,62
	RN		0,04	0,14	0,95	0,09	0,09
	decision		YES	YES	NO	YES	YES
test 10	sum	1,88	0,7	1,4	2,82	2,46	1,26
	RN		0,7	0,7	0,94	0,41	0,07
	decision		YES	YES	NO	NO	YES

Table 4: Shows the ranking of the views (test process "Prolima control") for 10 test cases from the developed formula. The threshold, random number and sum of each view for each test case can be seen. The important part is the outcome of the ranking seen as the decision in the table.

As can be seen in the table above (Table 4) the results of each test are influenced by the multiplication factor. It also can be seen that the used views for the merge can differ from one test to the next but what the results clearly state is that the first two views where taken in

every single test. From my point of view this shows that these two views are the nearest to the base process. In some cases as can be seen for view 5 which had never a chance to be picked because the sum lies over the threshold now was taken in five of ten cases. This shows that the factor of a random number can help to discover different ranking methods. With this method a great number of possible combinations can be generated but not the definite selection of views with the right factors. So using a random number in combination with formula 1 is not effective enough to be a merge logic by ranking.

Now formula 2 will be analysed in comparison to formula 1. The difference to formula one is now the use of the factor number of check words.

Formula 2		Line	View 1	View 2	View 3	View 4	View 5
test 1	sum	2,43	0,26	0	3,64	2,72	0,81
	RN		0,13	0	0,91	0,34	0,09
	decision		YES	YES	NO	NO	YES
test 2	sum	2,43	1,38	3,64	2,44	1,12	2,07
	RN		0,69	0,91	0,61	0,14	0,23
	decision		YES	NO	NO	YES	YES
test 3	sum	2,27	1,3	3,4	2,16	1,12	2,61
	RN		0,65	0,85	0,54	0,14	0,29
	decision		YES	NO	YES	YES	NO
test 4	sum	3,06	1,78	0,64	1,64	1,52	4,59
	RN		0,89	0,16	0,41	0,19	0,51
	decision		YES	YES	YES	YES	NO
test 5	sum	2,77	0,24	3,4	2,6	4,16	1,62
	RN		0,12	0,85	0,65	0,52	0,18
	decision		YES	NO	YES	NO	YES
test 6	sum	2,29	0,42	1,76	2,88	3,44	0,9
	RN		0,21	0,44	0,72	0,43	0,1
	decision		YES	YES	NO	NO	YES
test 7	sum	4,91	1,36	0,28	2,52	7,36	0,72
	RN		0,68	0,07	0,63	0,92	0,08
	decision		YES	YES	YES	NO	YES
test 8	sum	2,51	1,34	1,36	3,48	3,76	0,18
	RN		0,67	0,34	0,87	0,47	0,02
	decision		YES	YES	NO	NO	YES
test 9	sum	2,53	0,08	0,56	3,8	0,72	0,81
	RN		0,04	0,14	0,95	0,09	0,09
	decision		YES	YES	NO	YES	YES
test 10	sum	2,51	1,4	2,8	3,76	3,28	0,63
	RN		0,7	0,7	0,94	0,41	0,07
	decision		YES	NO	NO	NO	YES

Table 5: Shows the ranking of the views (test process "Prolima control") for 10 test cases from the developed formula with adaptation of the use of check words. The shown information are the same as can be seen in Table 4.



What can be seen is that in comparison to formula 1 the view 2 is not used in all ten tests therefore the variance is a bit higher also the number of NO's for view 5 has decreased. Therefore, the results look as constant as the results from formula 1. At this moment it can't be told if one formula is better than the other because the results lead both to a resulting merge graph. So in the following the other logic which uses a random number will be tested.

#### 4.2.2 Individual factors multiplied with random number

After this unsatisfying result from using a random number and multiplying it with the sum another method such as only multiplying the random number with single elements looked also useful. To decide which of the factors will be used for multiplying with the random number it was decided to take those which have the most impact on the result. As already explained in this paper the factors organisation level and number of sub processes looked like they have the most influence on the outcome therefore they were multiplied and not summarised in the formula. So these two factors will also be used to multiply them with the random number to get a different result.

$$\text{Formula1} = (MDOL * RN) * MDSP * ADDC + ADNP + (ADLL * RN) + ADNC$$

$$\text{Formula2} = (MDOL * RN) * MDSP * ADDC + ADNP + (ADLL * RN) + ADNC$$

Formula 1		Line	View 1	View 2	View 3	View 4	View 5
test 1	sum	1,95	1,536	0,52	1	0,24	0,7
	RN		0,11	0,52	1	0,24	0,7
	decision		YES	YES	NO	YES	YES
test 2	sum	11,84	1,7664	0,0864	0,56	1,766	0,1725
	RN		0,92	0,08	0,4	0,92	0,15
	decision		YES	YES	YES	NO	YES
test 3	sum	0,858	1,25	0,9881	0,1236	2,349	0,369
	RN		0,5	0,41	0,06	0,83	0,17
	decision		YES	YES	YES	NO	YES
test 4	sum	1,566	2,4396	4,2444	1,2876	1,388	3,9936
	RN		0,57	0,81	0,37	0,39	0,78
	decision		NO	YES	NO	YES	NO
test 5	sum	2,66	2,93	17,76	0,7416	0,45	3,677
	RN		0,22	0,99	0,06	0,04	0,27
	decision		NO	NO	YES	YES	NO

Table 6: Shows 5 of the generated scenarios for testing formula 1; random number multiplied to two individual factors.

After developing a few tests (more than 5 simulations were done but all showed an output similar to these 5 cases) it showed that the random factor increased and the difference between the results of the tests deviated more. The little bit of consistency from the first version of using random numbers disappeared which makes this approach worse than the last.

Formula 2		Line	View 1	View 2	View 3	View 4	View 5
test 1	sum	2,404	0,22	1,04	2	0,48	1,4
	RN		0,11	0,52	1	0,24	0,7
	decision		YES	YES	NO	YES	YES
test 2	sum	5,9	3,6060	0,2464	1,36	3,606	0,4725
	RN		0,92	0,08	0,4	0,92	0,15
	decision		NO	YES	NO	NO	YES
test 3	sum	1,352	1,75	1,398	0,1836	3,179	0,5389
	RN		0,5	0,41	0,06	0,83	0,17
	decision		YES	YES	YES	NO	YES
test 4	sum	2,404	3,58	5,8644	2,028	2,169	5,5536
	RN		0,57	0,81	0,37	0,39	0,78
	decision		NO	YES	NO	NO	NO
test 5	sum	3,7024	0,9504	8,851	0,2016	0,1296	1,2474
	RN		0,22	0,99	0,06	0,04	0,27
	decision		YES	YES	YES	YES	YES

Table 7: Shows 5 of the generated scenarios for testing formula 2; random number multiplied to two individual factors, one of the factors is number of check words.

The same conclusion goes for formula 2 but the effect of loss of constancy isn't as strong as for formula 1. However, with this result the approach is not usable. Due to the result of these tests the outcome wasn't what was expected therefore the graphs for formula 1 and 2 weren't drawn.

#### 4.2.3 Summary over the mathematical approaches

To summarize the development of a mathematical formula is to say that each approach lead to a merge that is an acceptable result but the one big problem each formula has is that when it is decided to use a view for the merge the whole change must be taken. It cannot be differed that one activity is going to be used and another is not. This should be possible in a merge therefore all these methods which were developed don't fulfil this requirement. A method where each element can be handled is to use an algorithm. So the next step includes developing an algorithm which uses the extracted factors to rank the views and differentiate between the elements.

#### 4.3 Developed Algorithm

As already explained the big problem of a mathematical formula is that in a view it can't be differentiated between activities which should be used for the merge and which should not. This case should be possible to appear therefore an algorithm will be developed. For this algorithm one base rule was decided that the algorithm should only use the selected factors but do so in every possible way. The restrictions and sequence from the developed formula can be ignored. The question now is how an algorithm could work with the selected factors? To find an answer every selected factor was analysed again.

#### Organisation level:

In the best case a list of organisation levels exists. The purpose of using this factor is to restrict the levels on the relevant one for the business process. To achieve this goal two things must be known. First the total number of levels and second the level where the relevant process is executed. Under the assumption that these two information are defined at the beginning of an algorithm a selection would be possible. Therefore, the restriction could be easily achieved. The total number of levels could be generated out of a system or defined by a human who counted out the hierarchical structure levels. If the total number is known the order of the levels must be defined. What does that mean? It must be defined if level number 1 is the top level or bottom level. When one level is fixated the rest can be automatically deduced and fixated. The number of the relevant level is difficult to find. In the worst case this level can't be found so the information is not available for the algorithm. What should be done if the relevant level number can't be defined? The solution which was chosen for this paper is that in this case the lowest level number will be used. This leads to the fact that no view will be excluded due to organisation level selection. For all following test processes the organisation level 1 will be the bottom level.

#### Limitation level:

The limitation level should reflect the rights of the actual person in comparison to other employees in the same department. After further inquiry in my old company the rights are specified with a special name and the assignment is worked off of a list. Therefore, it would never be possible to use the limitation name so another way of expressing the limitations must be found. The assumption for this factor is that the limitations are based on a department. Meaning, a department with 8 employees and three different types of employment (for example: employee with high responsibilities, employee, trainee) has also three different limitation levels. The people with higher responsibilities have more rights on programs and data than normal employees. Trainees will have fewer rights or at best the same amount as a normal employee. These three limitation levels are marking the limitation level for a department. In abstraction each other department can have the same employment types. Why should they have a different limitation level? The rights will be different because of the work they are responsible for but the level must not be different. Therefore, the limitation level represents an employment type. If the different types are specified with a number as an ID this number can be taken into the algorithm.

#### Number of sub processes:

Number of sub processes can be generated out of the view but to use it in an algorithm for an examination there must be some kind of mark able to define a value which can be used. To achieve this the already developed method of using a base as start point will be implemented into the algorithm. With this it is possible to use the factor number of sub processes for selection. But the difference alone isn't going to work because until now no negative values were taken into account. What would a negative value represent? It can't be said at this moment because a test period of more different processes would be needed to set the meaning of a positive and negative value. Furthermore, for the testing period different sizes of processes should be used.

#### Number of connectors:

Number of connectors is good for the selection but only if a number can be set which marks a point for comparison. Therefore, using a base value would fix this needed mark. Using this method is beneficial because both the value and marking value can be got out of the views. By using the difference of these two views a negative value could appear in the factor number of sub processes. A testing series on these factors would be best because this is difficult to accomplish. Therefore, the absolute value will be taken and negative values for this factor will be ignored in this paper.

#### Number of data connections:

Number of data connections is a structural factor like number of connectors or number of sub processes therefore the solution of using this factor is by using the method of a base view and ignoring negative values.

#### Number of involved people

Number of involved people is also a structural factor but this factor is also suitable for using the direct value instead of the difference between the base view and another view. What is better to use? The benefit of using only the number is that it could be used as a variable and set only once such as no more than 4 people for one process. Sounds good but it has a problem. What to do when 5 people are the base for a specific process? This would be difficult to implement into the logic therefore an adaptable frame must be used and by using the difference this could be accomplished. The difference, however, also needs a restriction. For example using 2 people for a process as the base in a view when there are 4 workers then it is no problem but if another person is needed for the approval of the outcome then a problem arises. Without a testing series the fixation of a frame is not possible for all processes so a frame will only be set for the test processes used farther in this paper.

#### **4.3.1 Description of the developed algorithm**

After completing this analysis one point was clear: the logic of using a base view must be implemented into the algorithm to guarantee that the structural factors can be compared (note that the designed algorithm is only a pseudo code and no real program code). Without such a comparison a ranking wouldn't be possible and therefore a selection of the views will only be made out of a comparison between the base and a specific view. To achieve the best result for merging the algorithm will be designed to only handle 2 views at the same time. One is the base and the second will be a different view. With this the algorithm must be executed several times to get a whole merged model but it is simpler to design an algorithm this way and furthermore it can be used on simple cases as well as on more complex processes. The possible cases which can appear such as insert or delete will be easier to handle when only one view is handled at the same time. Furthermore, the algorithm will be designed so that the changes will be implemented into the base process.

The process should be as in the following example; in the first round the base view and view A will be merged over the algorithm means. In the next round with base view and view B the changes from view A are already in the base view. The algorithm will be designed this way

because it brings the benefit that the changes such as insert will be found and a second insert will not be executed. A last step will be included. The analysis stated that the organisation level and limitation level could select the relevant area when the area is defined. By using my own experience for the programming of an algorithm I know that such comparison values are better out of the actual logic. With this in mind the frame of minimum and maximum for these two factors will be defined in variables.

After which logic is the algorithm designed? How does it work? These questions will be answered now. First the mentioned variables for organisation level (OL) and limitation level (LL) will be declared. They must be filled with a value even if it is a default value like 1 or 10. For example the organisation structure isn't known and the relevant area can't be defined. To avoid that the algorithm can't produce an output, low and high values should be declared. With this the variables are filled and every view will be further analysed on the other factors.

*Declaration variable maximum\_OL*

*Declaration variable minimum\_OL*

*Declaration variable maximum\_LL*

*Declaration variable minimum\_LL*

The next step included the first selection. It is best that with this selection most of the unnecessary views will be excluded. To achieve this, the factor with the best chance to do a restriction should be evaluated. From further investigations for the development of the mathematical formula it was clear that the organisation level and number of sub processes are the factors with the most influence. From my point of view - because of the knowledge of the hierarchical structure of a company - the factors organisation and limitation level are best suited to be the first selection criteria. After testing the organisation and limitation level as first selection criterion the outcome showed that both are good and the sequence is not important when both are used directly after another. So in the end the limitation level was chosen to be the first selection criterion and the organisation level as the second. Now an explanation of how each factor is implemented in the algorithm will follow.

#### Limitation level:

How did the logic for the limitation operate? It was analysed under two specific terms. Firstly: the rights are equal with the lowest rights level. This comparison was only made for complete evaluation of this factor because when the factor value is the same as the lowest level it means that the rights are too low because it marks only reading rights and no writing rights. In the worst case not even reading rights are assigned. Under the assumption that such a case is fulfilled no further investigation on this view will be made. If this comparison is not valid, in the next step a comparison is done to see if the value lies between a defined minimum and maximum. Is it true that the value lies in this area another investigation on the other factors will be made. Now what is the difference between the first and second comparison? Under the assumption that the worst case is present, there is no known structure and there are only default values for minimum and maximum then the difference is zero. If the minimum and maximum are defined then the difference of these two examinations lies only in the fact that the outcome is better documented. An evaluation of how many views were excluded because they have rights of the lowest level or of how many views were excluded because they are not

in the relevant area can be made. For the purpose of such evaluations these two comparisons were made. This is how it would look as a pseudo-code:

*When Limitation level from view equal with the lowest limitation level*

*Comment put to the console: "Views name" limitation level is the lowest limitation level*

*When Limitation level from view lays between the minimum and maximum inclusive the minimum and maximum value*

*Comment put to the console: next evaluation with organisation level*

*When Limitation level is outer the defined area*

*Comment put to the console: "Views name" not in defined area from limitation level*

#### Organisation level:

The only case which will be analysed in the algorithm is when the value ends up between the defined minimum and maximum. It is irrelevant if they are filled with default or real values. The default values makes no selection and every view will be send to the next evaluation of the next criterion. Why is there no other evaluation of this value? Simply because the default value takes all possible views and if the values are defined then only the relevant ones will be taken. This shows that no further comparison such as lower than minimum or higher than maximum is needed. To guarantee that every case is comprehensible and really in the defined area an extra check will be implemented that if a case appears that doesn't fall into the restricted area a comment will be put into the console that shows the relevant view. A possible scenario that can lead to such a need is a failure in the documented values which can appear if a human sets the factor values. In pseudo code this logic looks as followed:

*When Organisation level from view lays between the minimum and maximum inclusive the minimum and maximum value*

*Comment to the console: next evaluation with next factor*

*When Organisation level is outer the defined area*

*Comment put to the console: not in defined are from organisation level*

At this point in the algorithm the selection on the relevant views should be closed but only if the relevant variables are defined. Before going further in the evaluation of the criteria I want to give a brief overview of the current situation. Two different views are in the process of analysing the base view and another generated view. This other view should be ranked and after that merged into the base view. Now the problem appeared that if the ranking would be finished first the merge would be the same as the mathematical formula. Each change would be adopted and no difference could be made. To decide between the different changes is the goal for this algorithm. How can this be achieved? Through the solution which is included to mix the ranking and merging. The evaluation will be made and due to the outcome of it the merge will take place. With this scenario at this point the views must be checked on similarity. If this doesn't take place now during the different evaluations of the criteria every time a separate logic for checking similarity must be implemented. It is easier to take the logic out and only run through it once. This approach is made out of my personal experience with programming. The logic for finding similarity will not be defined because many different approaches already exist. [5] One of these can be taken and implemented at this point in the pseudo code.

*Run through the base process and at the same time through the relevant view*

*Check if the current activity is equal in both views*

*Comment to the console: "Activity Name" exists in both views (base view name / view name)*

*When both are not equal*

*Comment to the console: next evaluation with next factor*

#### Number of connectors:

Now that it is confirmed that two activities are not the same the next step includes the examination if this activity is part of a bigger complex. That means that it must be found out if the activity is really a connector and in the following more activities will be inserted too or if it is only one single activity which should be inserted. To achieve this, the developed system of using the difference between the base view and the current view will be used but because the difference also can be negative the absolute value of the difference will be used. If the difference is zero than the actual non-equal activity is no connector so the deduction that it is a single activity can be made. Therefore, the activity can be inserted and if the difference isn't zero a further investigation with the next factor will decide what to do with this activity. After inserting this activity when the difference is null it must be controlled if a data set must be inserted or not. Therefore, the next factor Number of data sets must be controlled. The pseudo code for this factor looks like the following:

*When the absolute value of the difference between base view and actual view is equal to zero*

*insert activity that is not equal*

*When the absolute value of the difference is not zero*

*Comment to the console: next evaluation with next factor*

#### Number of data sets:

This is important for the merge to guarantee that the data sets are also inserted into the merged model. Therefore, it must be controlled if the actual activity which should be inserted brings an extra data set or not. Now the difference of number of data sets will be analysed. If it is zero then no new data sets exists and the insertion is completed. If the number is greater than zero, a logic must be implemented that checks if the extra data set is connected to the actual activity. This has nothing to do with the ranking only the logic to follow the data link takes place because data sets are in every system and the graph is saved in different types. Only the hint that it must be controlled will be given. If a data link is found then both the activity and data set must be included. Furthermore, it must be controlled if the new data set has another connection to an activity. This activity can be found during a set link but only if this information is saved in the data sets options. If this kind of information is saved then the other activities must be inserted too because when they don't exist it should be included when the other activities will be inserted. If the link goes to an activity that already exists then with this logic the link will not be found so the responsibility of fulfilling this information belongs to the person who looks over the finished merged model. Now an example of how such a logic could look like as pseudo code will follow:

*Logic for checking the data objects:*

*follow the link set in the activity to the data object*

*when the data object already exists insert the link*

*when the data object doesn't exist insert the link and data object*

*look through all data connections from this data object*

*if there are other links follow each link*

*look for every link if the activity to which the link goes already exists*

*when not insert it too*

*when it exists insert the link*

After developing this far it appeared that during the factor number of connectors the factor for data sets was completely ignored. This means that when inserting an activity at this point the data sets were not included. So an adaptation must be made that the same logic as explained in this factor will be executed after inserting the activity. The pseudo code for the adapted factor number of connectors and the following evaluation of the factor number of data sets looks like following:

*When the absolute value (number of connectors) of the difference between base view and actual view is equal to zero*

*insert activity that is not equal*

*execute logic for checking the data objects*

*When the absolute value (number of connectors) of the difference is not zero*

*Comment to the console: next evaluation with next factor*

*When the absolute value (number of data connections) of the difference between base view and actual views is equal to zero*

*insert activity that is not equal*

*When the absolute value (number of data connections) of the difference is not zero*

*execute logic for checking the data objects*

The factor number of data connections isn't finished yet. At the moment this factor will only be analysed when a difference between the number of connectors exists. What if there isn't a difference but an activity with a data object? This logic doesn't execute this case therefore in the next round this possible way will be implemented by using the logic mentioned above again.

The next step included finding out what to do when both factors number of connectors and number of data connections aren't different. This is important because the insertion of the activity only happens when one of the two factors are different to the base. What can it mean when none of these factors appear? Still a normal insertion but there is also a possibility that it is not a normal insertion but a position change of an existing activity. At this moment no method was found to find out, if it is a change or not except the way of using a loop to control the base process if this activity appears in a later position. Nevertheless, in both cases the activity will be inserted so the case of position changing isn't implemented into this algorithm. If it is really a position change then the procedure will be processed over an insertion and deletion of the same object on two different positions.

Now in the next step the deletion will be processed. Up until now in the process, insertions were made so now the view falls under no factor which can show that an extra activity exists. The present point of knowledge is that it could be possible that the current activity is a



deletion but also an insertion. When an activity is deleted and another inserted it can't be processed in the algorithm so far. What needs to be done to find out if it is a deletion or insertion? The way that's used is to see through one whole process in this case the base process because all changes will be implemented into it and see if the needed activity is on another position or not. If this activity is found, an insertion will be made when a deletion isn't noted. The deletion will take place later and only the activity will be marked as a deletion. With this method further insertions of other activities should be guaranteed because both the predecessor and follower are present.

*Run through the base process and check every activity for equality*

*When an equal activity is found*

*insert activity that is not equal*

*When no equal activity could be found*

*Mark the current activity as deletion*

Now the point is arrived where the functions insertion and position change were identified and processed. The only remaining function is the deletion. It appears that at this point of the logic it means that the current activity must be deleted because it is no insertion or position change. Should every deletion be executed? The answer is no because a deletion can have different reasons and one reason is that the rights of the person from the current view are too low to execute this activity. After all, the merged model should be near the real life process therefore a restriction was set onto the deletions. Only the deletions from views with an organisation level as high as the base view or one level lower should be executed. With this restriction the problem of too low rights should be avoided.

*When difference between organisation level from base view and actual view smaller or equal the value one*

*delete activity that is not equal*

*When difference between organisation level from base view and actual view greater than one*

*Comment to the console: organisation level too low for deletion*

Now every function for the activities are processed and the data object with their data connections also are processed. What remains? Two factors are not used until now: the factor number of sub processes and number of involved people. How can they be adapted to this logic? All possible functions in combination with data sets are processed so a further adaptation is not needed but a search for a way of using these factors was performed. The factor number of sub processes could be implemented before controlling the number of data connections. The number of sub processes can also help as much as the number of connectors to identify a possible other way for the insert. After finding a place for this factor the problem was how to process a sub process. The only way was to execute a sub process the same as if it were a normal activity to find out if it has a data object which is new, has other links to this data object and how to insert it as a normal insertion or a position change. This scenario led to the fact to not use this factor in the algorithm but process a sub process the same way as an activity. Next is the factor number of involved people which is also a good selecting factor at the beginning. It could be used between the organisation level and number of connectors. This is a difficult factor to restrict because the number of involved people could be higher if the process itself is greater than the used ones in this master thesis. To provide different types of greater processes with more involved people is rather difficult. Therefore,

the point of adapting this factor will be restricted onto the test processes used in the later part of this master thesis. Under the assumption that a lower right includes the cooperation with a second person the limit of this factor will be set by a difference from two people. Why a difference of two? The fact that a third person will be needed for confirming the outcome of the process is included into this value. This logic looks like the following in pseudo code:

*When the absolute value of the difference between base view and actual view is smaller or equal than three*

*Comment to the console: next evaluation with next factor*

*When the absolute value of the difference between base view and actual view is greater than three*

*Comment to the console: "Views name" too many involved people*

### 4.3.2 Summary of the developed algorithm

Now the whole algorithm will be shown:

*Declaration variable maximum\_OL*

*Declaration variable minimum\_OL*

*Declaration variable maximum\_LL*

*Declaration variable minimum\_LL*

*When Limitation level from view equal with the lowest limitation level*

*Comment put to the console: "Views name" limitation level is the lowest limitation level*

*When Limitation level from view lays between the minimum and maximum inclusive the minimum and maximum value*

*Comment put to the console: next evaluation with organisation level*

*When Organisation level from view lays between the minimum and maximum inclusive the minimum and maximum value*

*Comment to the console: next evaluation with next factor*

*Run through the base process and at the same time through the relevant view*

*Check if the current activity is equal in both views*

*Comment to the console: "Activity Name" exists in both views  
(base view name / view name)*

*When both are not equal*

*Comment to the console: next evaluation with next factor*

*When the absolute value (number of connectors) of the difference between base view and actual view is equal to zero*

*insert activity that is not equal*

*execute logic for checking the data objects*

*When the absolute value (number of connectors) of the difference is not zero*

*Comment to the console: next evaluation with next factor*

*When the absolute value (number of data connections) of the difference between base view and actual views is equal to zero*

*insert activity that is not equal*

*When the absolute value (number of data connections) of the difference is not zero*

*execute logic for checking the data objects*

*When the absolute value (number of data connections) of the difference between base view and actual views is equal to zero*

*insert activity that is not equal*  
*When the absolute value (number of data connections) of the*  
*difference is not zero*  
  
*execute logic for checking the data objects*  
*Run through the base process and check every activity for*  
*equality*  
  
*When an equal activity is found*  
*insert activity that is not equal*  
*When no equal activity could be found*  
*Mark the current activity as deletion*  
*When difference between organisation level from base view and*  
*actual view smaller or equal the value one*  
*delete activity that is not equal*  
*When difference between organisation level from base view and*  
*actual view greater than one*  
*Comment to the console: organisation level too low for*  
*deletion*  
*When both activities are equal*  
*Comment to the console: both activities equal*  
*When Organisation level is outer the defined area*  
*Comment put to the console: not in defined area from organisation level*  
*When Limitation level is outer the defined area*  
*Comment put to the console: "Views name" not in defined area from limitation level*

To summarize the functionality of the developed algorithm: out of the six selected factors for the mathematical formula, 5 were used. 3 of them were used for selecting the views the other two help to differentiate between the change functions insertion, position change and deletion. The selection process is based on two steps. In the first step the two factors organisation level and limitation level should be restricted on an area in the organisation hierarchy. If the relevant area for the process isn't known default values should be inserted such as the lowest and highest level number of the hierarchy. After this first selection a second one will be processed with the number of involved people. This restriction is merely based on the used test processes in the further work. It dictates that only 3 people should be involved in one process. This evaluation is based on the idea that when two people are working on one process maybe one further person is needed for the approval. After selecting the views the algorithm deals with the function insertion, whereat the data connections and data objects are also included. This means that when a new activity should be inserted and it needs a new data object this will also be implemented. If this new data object is also needed from other activities these links will also be set in the algorithm. For the insertion of activities two different approaches were followed. First if a difference in the number of connections exists it will be tested and in a further part the data links are tested. If there is no difference the second path will be processed where only the data objects are further analysed. In both ways the other one is excluded so only one insertion takes place. If both are not valid it is tested if the activity is part of a position change, when it is it will be processed over an insertion and deletion on the two positions. If not the activity must be deleted which will only be executed if the organisation level is only one level different to the base process.

## 5 Testing on case study

In this chapter the created algorithm (chapter 4) with the selected factors (chapter 3) will be tested. The testing is set on three different processes and branches out to see if it is usable on more than one process and in more than one branch. The procedure for the three test cases will be the same when the process is introduced and the description will be based on the defined original process. In the first case “Prolima control” the process which is currently used by the company Raiffeisenlandesbank Niederösterreich-Wien will be used. The process was provided for this master thesis. In the other two cases the processes are based on tasks of the exercise sheet two from the university course Workflow Technologies from the winter semester 2013/2014 at the University of Vienna. After introducing the used process the scenario which is used to simulate the different views is explained. The scenario for test case one is a real life scenario and the other two scenarios were designed by me. The self-designed scenarios are based on real life situations I experienced myself and which are now used to simulate a department, the individual people and the views. After describing the basic construction the different views will be introduced shortly by describing the differences between the original process and the view. Now the algorithm is applied to the views and a merge is generated. Therefore, the algorithm can handle only two views at a time where one is the base view and the other one is a view which should be merged into the base so the merge has to be done multiple times. It must be noted that in the algorithm logic the base view is changed because the changes of the other views are executed on the base. This means that during the first run between the base and a view the base process will be adapted and in the second run the base view is a different one than in the first run. Furthermore, the used limitation level, organisation structure and the point table must be known which will be explained shortly. In a final step the generated merge model will be explained by describing what structural parts are changed compared to the base view. For better traceability the name of the view from where the change is will be written over the marked change.

### 5.1 Testing case Prolima control

In this chapter the test case process “Prolima control” will be tested. To summarize the procedure: At first an introduction about the process itself will be given and after that the real life scenario will be explained with the corresponding views which are generated out of interviews with the people of the real life department in the company Raiffeisenlandesbank Niederösterreich-Wien. Lastly, the merged model which was generated by using the developed algorithm is explained.

#### 5.1.1 Introduction of the process

In reality the prolima control process is a part of the implementation process of a product into the company. This special prolima list only holds information about bonds. Bonds can be handled on different markets and appear with individual specifications. An example would be that a bond is handled in the currency Russian ruble. This bond wouldn't be in the catalogue but if the bond is adapted to Euro this bond would show in the catalogue. This means that variations of bonds can be made individually in a bank but it must follow a specific process so that the bond is registered, sent to the management and is accepted from this stage. It is essential that it is accepted because each bond holds a risk factor and this risk must be calculated and evaluated from the management because a decision must be made based on the result of this factor. If the company can take the risk the variation can be handled but if the

risk can't be handled no business with this bond variation may be done. Furthermore, this variation must fit into an already existing rating system from a bond.

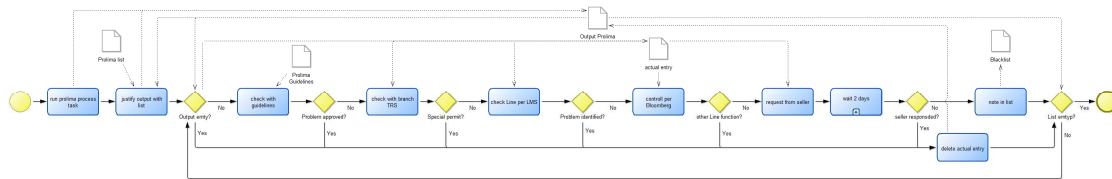


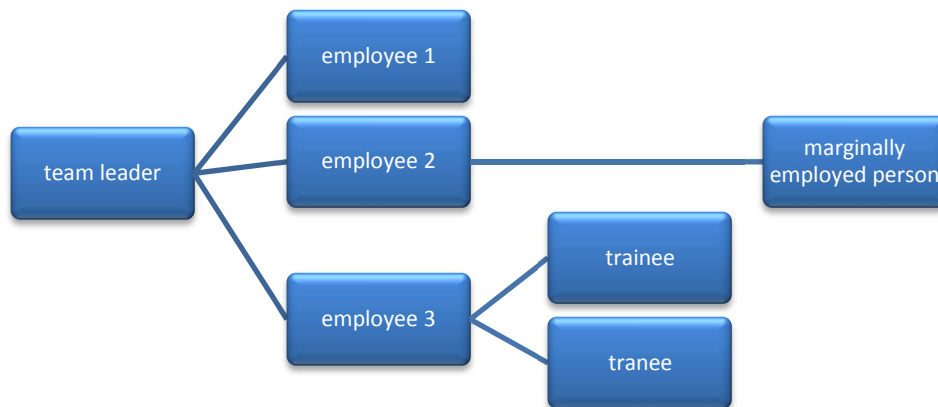
Figure 2: This picture shows the original process which is used from the company Raiffeisenlandesbank Niederösterreich-Wien. It was provided for this master thesis. A bigger picture can be found in appendix 12.2.

Part of this process is a program which first runs through all handled business from the day before the actual and saves the new or adapted deals in a list. The first step of the shown process is the execution of this program. After that in a second step the generated list is compared to the deals in the proloma catalogue. The outcome of this comparison is that all from the management granted business is taken out of the list and only those remain which are not allowed. If an entry is left it must undergo the following procedure to figure out if the deal is legal because it is a further adaption of an approved deal or not. Next step in the process is an or-gate where it is checked if the remaining output is empty. In this case the process ends but if the outcome is not empty each found deal will be executed separately in a backwards loop. In the following description “deal” refers to a single business deal because the procedure has to be executed for each found deal in the list. The first step in the loop is to refer to the handbook or other guidelines to look up in which case the deal can be put into. When a suitable case is found the deal is legal so an evaluation of the found information is done. If the problem is then solved the process goes to the end of the loop but if it isn’t solved the next three steps use different tools to take another look at the deal information. After each tool was checked, an evaluation of the found information is done and if the problem is solved this way, the end of the loop is reached. However, if the problem persists certain guidelines are saved to deal with the problem at a later time. Under the assumption that at this point of the process no solution could be found in the further information of the deal the seller of this deal is contacted via mail. After two days if an explanation of the seller comes back it will be added to the deal in an extra list and the case ends at the loops' termination. If no reply comes back, the deal will be listed in a report for the management to solve it there and then the end of the loop is reached. In every case the end of the loop is reached when the output isn't empty which means one case of the output is solved but when other entries exists they must also be solved. At this point it will be examined if another entry exists or if the loop starts again until no entry is left. Only after all deals are executed the end of the process is reached.

### 5.1.2 Department scenario

After explaining the process the scenario of the department will be explained. This is important because the simulation will only take place in the relevant department and not the whole company. The reason why only the department and not the whole company is chosen is because I worked in the department and therefore know the circumstances of which people are involved in the process and who executes it and who doesn't. This knowledge is used to simulate the correct situation of the department for the test in a level structure. A model of

the group was made (figure 3) to show the connections between the levels of the individual people in this department. Each simulated participant in this model is based on a real person of the department I worked in.



*Figure 3: This picture shows the connections between the people in the department and their level for the test process proloma control. The level structure is based on the individual organisation level.*

As can be seen in the diagram, the department is made up of 4 levels. The lowest level is the one I was in at that time this department picture was created. My employment was only that of a marginally employed person with only 8 working hours a week. Therefore, it was easier for me not to have the rights for different working programs because a licence costs and the limitations can reach such a high level that it wouldn't be adequate for a temporary worker because of security issues. It is better if another person with a licence delivers the relevant information or is helping in the process. This can not only be used for this level but also for the next one which holds 2 people who are both trainees. As a trainee for a period of 6 months their job is to work like a regular member of this department. Therefore, they need appropriate rights on systems but also enough restrictions so that sensible data is not handled alone. The control of this level is made from the next level which consists of the permanently employed people. Level 3 is divided into two different cases of limitation because from these 3 people one of them (employee 3) has more responsibility than the other two. The reason for that is that this person is the substitute of the team leader and therefore he must have the ability to do all the jobs the leader has while including the appropriate limitation level for example the handling of the budget. Atop of this model stands the team leader which holds the responsibility for all the work from this department plus he has all rights on data and programs. Furthermore, he must also deliver part of the work and must be able to control the work processed from the other people in the department.

### **5.1.3 Individual views and the positions in the scenario**

The used scenario is taken out of a real company therefore the views of the people in the scenario are all existing and have a real view. How did I get the different views of the people? One of them is the process I executed myself so this view could be got easily but for the other views each person of the department was interviewed about their work. The interview was held in German and the base of it was a guideline constructed for the purpose of finding all relevant information for the process. This guideline can be found in the appendix 12.1. Afterwards the gathered information was designed into views. In the following every

generated view will be explained shortly. The pictures of the views are small so larger images can be found in the appendix 12.2. What will be explained in the following? The original process was described in a previous part of this chapter therefore only the differences between the individual views to this original will be explained. To show the differences in a better fashion they will be described per text, in a formal way after [9] and also per marking the differences with colours in the view. Three different colours are used where red stands for deletion, green symbolises an insertion and blue represents a change of the position from an activity.

#### Team leader:

The team leader holds not only job responsibilities but also organisational ones. He has the highest level of rights in the department but this doesn't mean that he's the only one. The appropriate rights for the department are also given to other employees. The differences are in the duties and responsibilities.

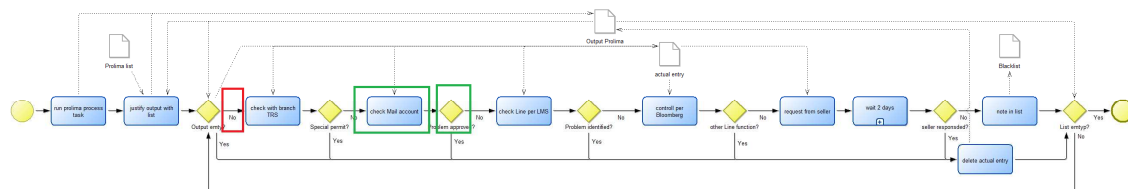


Figure 4: Shows the view of the team leader of the group who is responsible for this process. This view will be used as the base view for the algorithm.

What changed in comparison to the original process is the deletion of an activity named “check with guidelines” and the associated data object. The reason why this person doesn't need to look in the guidelines is because this person wrote the document and is also responsible to keep it updated. Therefore, the knowledge is enough to execute this activity automatically during the further process and doesn't need an extra point in the process diagram. The second change is the insertion of an activity and a connector. The reason for this insertion is that a dealer often sends some information about the dealt object to the department for example the got approve from the management. This information can be the cause of not passing the evaluation. So in the activity “check mail account” this scenario is checked and after that an evaluation is made over a connector if the case is solved or not.

*Data Object: delete (team leader, Prolima guidelines)*

*Activity: delete (team leader, check with guidelines)*

*Activity: sInsert (team leader, check Mail account, Special permit?, check Line per LMS)*

*Connector: sInsert (team leader, Problem approved?, check Mail account, check Line per LMS)*

Employee 1 holds the responsibility of representing the team leader therefore he has the same rights and also the greatest responsibility if the team leader isn't available.

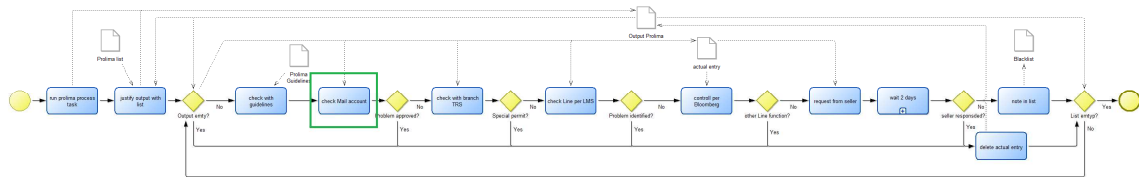


Figure 5: Shows the individual view of employee 1 on the scenario.

The only appearing change in this case is the insertion of the already explained activity “check mail account”.

*Activity: sInsert (employee 1, check mail account, check with guidelines, Problem approved?)*

Employee 2

Employee 2 is on the same level as employee 1 but doesn't have the same rights. In this scenario the program Bloomberg is not assigned to the employee because the license costs for this program are too high so for monetary reasons the license costs were cut.

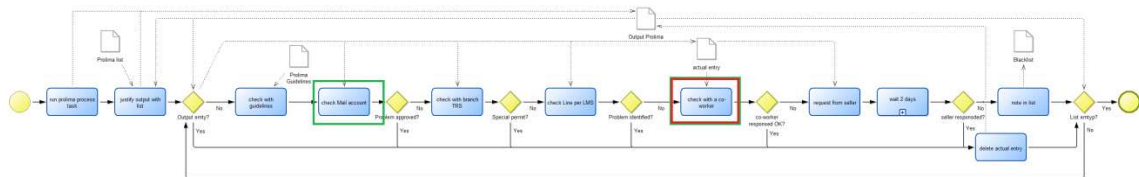


Figure 6: Shows the view of employee 2 from the test process “Prolima control” scenario.

The first change is the addition of an activity called “check mail account”. This activity is also inserted in other views to check for information which could be delivered from the dealers. Now that the program Bloomberg isn’t available this activity is deleted from the process. At the same time another activity replaces the position which is called “check with a co-worker”. This activity is the substitute for the Bloomberg program because the fact that a check with this program is still needed therefore this person must ask a colleague of his to look into Bloomberg and give him the results. The interaction with the co-worker wasn’t modelled but the activity should represent it.

*Activity: sInsert (employee 2, check mail account, check with guidelines, Problem approved?)*

*Activity: replace (employee 2, control per bloomberg, check with a co-worker)*

## Employee 3

In the case of employee 3 the tasks are divided into two parts where this employee and one other with higher rights execute the process together. This means that employee 3 doesn't have the appropriate rights for this process which leads to a process diagram with more activities. They are needed to compensate the handover of the data from one person to the other. The communication between these two isn't designed. The handover of the data



symbolises that a communication and flow of information between the two involved people isn't planned.

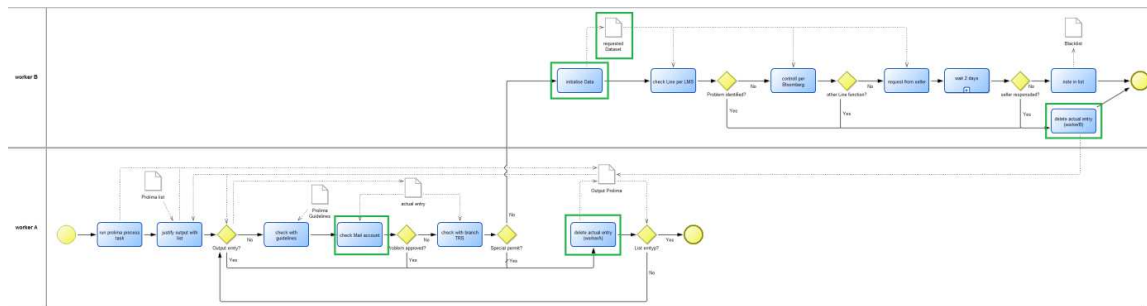


Figure 7: Shows the whole process of the view from employee 3. The lower part is executed from employee 3 and the upper part needs the aid of a colleague with higher rights.

The first insertion is the activity “check mail account” which is already explained because it appears in other views too. In the next step an insertion of the activity “initialise data” is performed which represents the handover of the gathered information from employee 3 to a colleague. The other two insertions are only an adaptation of an existing activity. “Delete actual entry” appears now two times because there are two separate working paths and depending on which part the solution is found in this part the entry of the list is being deleted. Because of the policy of the used tool each name of an activity must be unique therefore the same process couldn’t appear a second time. The old activity is deleted and two new activities are created. The names of the new activities are “Delete actual entry (worker A)” and “Delete actual entry (worker B)”. With this the naming conventions were satisfied but it is obvious that it is the same activity only from different workers performed.

Activity: *insert (employee 3, check mail account, check with guidelines, Problem approved?)*

Activity: *insert (employee 3, initialise data, Special permit?, check Line per LMS)*

Activity: *delete (employee 3, delete actual entry)*

Activity: *insert (employee 3, delete actual entry (worker A), Special permit?, List empty?)*

Activity: *insert (employee 3, delete actual entry (worker B), Seller responded?, end event)*

### Trainee:

Often six month work experiences are given out and the chosen person needs to do daily business. To achieve this, appropriate rights for the tasks are needed but often licence software is not part of the rights because it is too expensive. In the following case an extra workstation with the licence for specific software exists where each person of the department can work. So the trainee is able to use that extra work station.

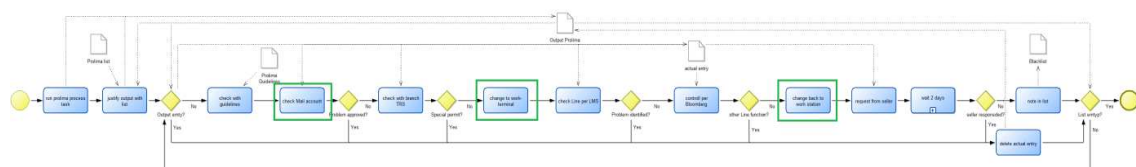


Figure 8: Shows the view of the two trainees from the generated scenario. Both have the same limitation and organisation level the view is exactly the same.

In this example the already described activity for “checking mail account” is inserted. The other two insertions are part of the scenario for using a different work station. These two activities are inserted to represent the act of changing the physical work place position.

Activity: *sInsert (Trainee, check mail account, check with guidelines, Problem approved?)*

Activity: *sInsert (Trainee, change to work-terminal, Special permit?, check Line per LMS)*

Activity: *sInsert (Trainee, change to work-terminal, other Line function?, request from seller)*

### Marginally employed person

A marginally employed person only works a few hours a week and because the time to teach this person work activities and the used programs is too short so only supporting activities are possible. In this case no rights for programs are assigned to the marginally employed person.

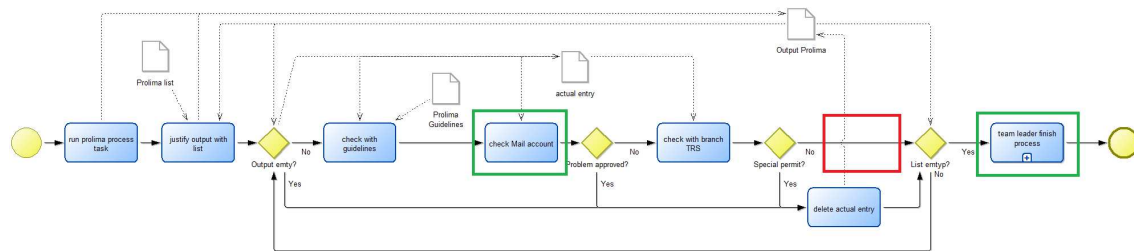


Figure 9: Shows the view of the marginally employed person.

The right to look in the mail account is present so a new activity “check mail account” can be inserted. Nevertheless, there are no other rights so every check over programs is being deleted along the following activities where the dealer is contacted. To compensate the loss of the rights a sub process is inserted which is named “team leader finish process”. This sub process means that the work of checking the programs is handed over to another person, in this case the team leader. The sub process holds a logic other than the described view of the team leader because a part of the process is already executed.

Activity: *sInsert (marginally employed person, check mail account, check with guidelines, Problem approved?)*

Activity: *delete (marginally employed person, check Line per LMS)*

Connector: *delete (marginally employed person, Problem identified?)*

Activity: *delete (marginally employed person, control per bloomberg)*

Connector: *delete (marginally employed person, other Line function?)*

Activity: *delete (marginally employed person, request from seller)*

Activity: *delete (marginally employed person, wait 2 days)*

Connector: *delete (marginally employed person, seller responded?)*

Activity: *delete (marginally employed person, note in list)*

Data Object: *delete (marginally employed person, Blacklist)*

Sub process: *sInsert (team leader finish process, List empty?, End Note)*

### 5.1.4 Merged model

Before describing the output of the used algorithm the base levels will be explained. In the following two tables the levels and a description are shown for the limitation level and organisation level. This information is used in the algorithm because this example is a real life case the organisation level is taken from a real scenario compared to the limitation level which isn't. The limitation structure of the relevant department could not be identified therefore this

is a self-constructed level structure for rights. In the execution of the algorithm the limitation level wasn't restricted because only the relevant levels are known. Through my personal experience and the unavailable views from the irrelevant levels the organisation level was restricted to the values of one to four.

Limitation level	
1	visitor
2	employee2
3	employee1
4	team leader

*Table 8.1: Shows the defined limitation level for the test process Prolima control. This table is based on personal experience.*

Organisation level	
1	marginally employed person
2	trainee
3	employee
4	team leader
5	department leader
6	head department leader
7	managing committee

*Table 8.2: Shows the defined organisation level for the test process Prolima control. This table is based on personal experience.*

The limitation level (table 8.1) and organisation level (table 8.2) are the last pieces which were needed to create the point table. The point table is the starting point of the algorithm. If this table is non-existent or incorrect the outcome of the algorithm can't be produced or is incomplete in the best case. So the defined level structure and the other gathered factors out of the views were collected in table 8.3.

	# sub processes	Organisation level	# data connections	# connectors	# involved people	Limitation level
<b>team leader</b>	0	4	14	7	1	4
<b>employee 1</b>	0	3	15	7	1	4
<b>employee 2</b>	0	3	15	7	2	3
<b>employee 3</b>	0	3	16	7	1	3
<b>trainee</b>	0	2	15	7	1	2
<b>marginally employed person</b>	1	1	12	4	1	2

*Table 8.3: Shows the finished point table for the test case "Prolima control". The organisation level and limitation level are based on table 8.1 and 8.2. The other factors were counted out of the views.*

After defining the used point table for the algorithm a merged model was generated by executing the developed algorithm. In the following the model will be described or more precisely, the differences between the base view and the finished merge model will be explained in text form and after [9]. In figure 10 not only the differences are shown through colour coding (same meanings as in a previous part of this chapter) but also the view is named from the origin of the individual changes. Due to the fact that the picture of the model is small the details are not readable but a bigger picture can be found in the appendix 12.2.

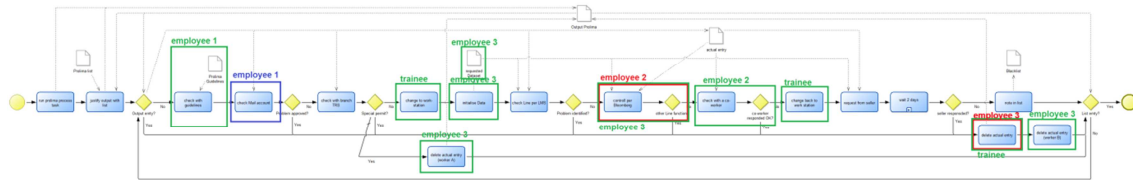


Figure 10: Shows the generated merged model for the test case “Prolima control”. The changes are pointed out per colour and the view which is responsible for the change is put over the coloured markings.

First of all, the order in which the single views were taken into the merged model is employee 1, employee 2, employee 3, trainee and lastly the marginally employed person. This means that the description of the inserted changes will be following the order the views were taken into the merge. The changes from employee 1 include the insertion of the activity “check with guidelines” and “check mail account”. The activity for the guidelines and the corresponding data object are inserted because they don’t exist in the model. Furthermore, the activity for mail checking already exists in the base model but on a different position so a position change to the new position will be executed because the organisation level has the appropriate level. Next the changes from view employee 2 are executed. Employee 2 includes 1 change on the process which adds the deletion of the activity “control per Bloomberg” and inserts the activity “check with a co-worker” as a substitute. The following or-gates for evaluating the result are also deleted and inserted to match the activity. Due to the organisation level which is one step lower than the base view the deletion is executed and the other activity is inserted.

The next step consists out of merging the changes from the view of employee 3 to the base view. Most important is that the previously deleted activity for Bloomberg is now inserted again. The next change affects the activity “initialise data” which is used for communication and interaction between two involved people. This activity is inserted because the organisation level is appropriate and the activity doesn’t exist in the base view. Furthermore, a corresponding data object named “requested Dataset” must be inserted with all connections to other activities. The last change is needed because two people are involved. In the used tool the naming of an activity must be unique therefore in this view the same activity exists a second time with a slight change in the naming. The division of the same activity is inserted in the base process while the old activity “delete actual entry” is deleted because the organisation level of this view is maximum one step lower than the base process. Next is the view of the two trainees. The removed activity for deleting the actual entry is once more inserted into the model because it is a change between the base and the view of the trainees. The other two changes are the insertion of the activities “change to work terminal” and “change back to work station”. Lastly the changes for the view marginally employed person aren’t implemented because the limitation level is the lowest which excludes the view from using.

Activity: *sInsert (merge Prolima, check with guidelines, Output empty?, Problem approved)*

Data Object: *sInsert (merge Prolima, Prolima Guidelines, , check with guidelines)*

Activity:

Activity: *sInsert (merge Prolima, change to work station, Special permit?, check Line per LMS)*

Activity: *sInsert (merge Prolima, initialise Data, change to work station, check Line per LMS)*

Activity: *delete (merge Prolima, control per bloomberg)*

*Connector: delete (merge Prolima, other Line function?)*

*Activity: sInsert (merge Prolima, control per bloomberg, Problem identified?, request from seller)*

*Connector: sInsert (merge Prolima, other Line function?, control per bloomberg, request from seller)*

*Activity: sInsert (merge Prolima, check with a co-worker, other Line function?, request from seller)*

*Connector: sInsert (merge Prolima, co-worker responded OK?, check with a co-worker, request from seller)*

*Activity: sInsert (merge Prolima, change back to work station, co-worker responded OK?, request from seller)*

*Activity: sInsert (merge Prolima, delete actual entry (worker A), Special permit?, List empty?)*

*Activity: delete (merge Prolima, delete actual entry)*

*Activity: sInsert (merge Prolima, delete actual entry, seller responded?, List empty?)*

*Activity: sInsert (merge Prolima, delete actual entry (worker B), delete actual entry, List empty?)*

## 5.2 Testing case university

This chapter shows the testing of the process “noting from a university course”. The procedure for the testing is the same as test example one Prolima control. The structure of this chapter is that first the used process will be introduced. Note that the source of this process is a task from an exercise sheet of a university course so only one process is known therefore this view is set as the original process. After that the self-designed scenario for the department will be introduced with the corresponding views which are made by me. All generated views are based on the original process from the task. Lastly the merged model will be explained.

### 5.2.1 Introduction of the process

The process for this test case deals with the procedure of noting but not for a school just a university. The difference lies in the fact that first of all a different law (in Austria UnivG for university and SchUG for school) is effective and second the internal structure between a university and school differs. What effect can the structure have on the noting process? In a school a teacher for a specific subject is assigned to a number of classes. Normally this teacher and only this teacher is responsible for the noting but what happens if during the school year this teacher gets sick. Two different methods can be used in such a case. First the subject is cancelled or second another teacher from the same subject substitutes. The first case doesn't change the fact that the teacher is responsible for noting but in the second case the substitute teacher holds the duty to document the taught material and the behaviour of the students. If a student's behaviour is really bad and the substitute teacher reports this to the responsible teacher, he should use this information for the noting. This means that in scenario two teachers are responsible for the noting but the originally assigned teacher makes the last decision. So now the difference to a university lies in the fact that more parts than a single professor can be involved with one course. These people are lower in their academic title such as an ongoing doctor or a doctor who doesn't yet hold the right to teach alone. Such people need a professor who works as the responsible teacher but in reality they don't teach or they don't do it alone. Furthermore, the concept of a tutor mustn't be neglected. After introducing the point of departure, now the original process for the test will be introduced. The source for this test process is a task from an exercise sheet of the course Workflow Technologies (WS 13/14) from the University of Vienna. The task is described as follows:

„Zunächst gibt der Benutzer die Note der Klausur ein. Anschließend gibt er dann in einer Schleife die bei den einzelnen Übungsblätter - sofern bearbeitet - erreichte Punktzahl sowie die jeweils maximal erreichbare Punktzahl ein. Dies geschieht so lange, bis alle bearbeiteten Blätter erfasst wurden. Insgesamt seien 15 Übungsblätter ausgegeben worden, d.h. der Benutzer soll bis maximal 15 Übungsblätter entsprechende Eingaben tätigen können. Aus diesen Informationen wird im Prozess die endgültige Note berechnet und diese zum Schluss dem Benutzer angezeigt. Bei der Berechnung der endgültigen Note sollen folgende Regeln gelten:

- Hat der Teilnehmer in der Klausur bereits die Note 1,0 so ändert sich diese nicht.
- Hat der Teilnehmer mindestens 2/3 der Blätter abgegeben und auf den abgegebenen Übungsblättern im Durchschnitt mehr als 60% der erreichbaren Übungspunkte bekommen, so ist die endgültige Note um 0,3 besser als die Note der Klausur. „

Source: Rinderle-Ma S., Course Workflow Technologies, exercise sheet 2 exercise 2.3 from winter semester 2013/2014, University of Vienna

This task was taken as a basis and the original process was designed after it. The finished design can be seen in the following as a small picture (figure 11). The same process as a bigger picture can be found in the appendix 12.3.

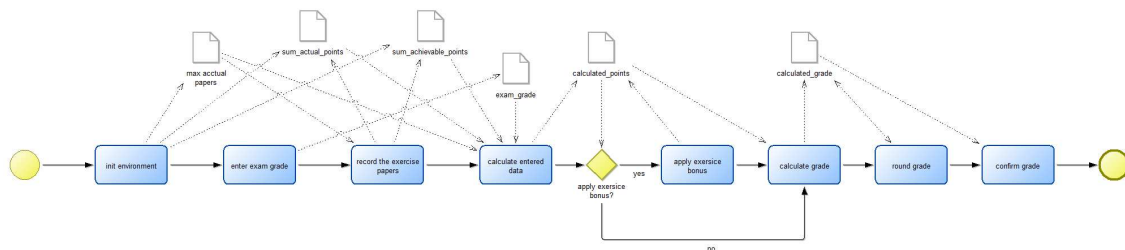


Figure 11: Shows the designed process based on the task from the exercise sheet of the university course Workflow Technologies. Changes from the text to the design were executed to make the original process simpler in structure.

First of all a starting object is generated and after that follows the first activity which process is the initialisation of the further used variables. These variables are the number of exercise sheets, number of total points which can be got on the sheets and the number of accomplished points. The next activity implements the grade of the final test. After defining these variables the next step is to calculate a total grade based on the points and grades the student received. Now in the next step is a decision which questions if an exercise bonus is used or not. If “yes” the points will be summarized with the calculated points and if “no” the process goes on without any change to the grade. Now the calculated points will be examined over a grading key to get the grade. The next step is for rounding the grade to a natural number. Lastly the grade must be confirmed which leads to the end of the process.

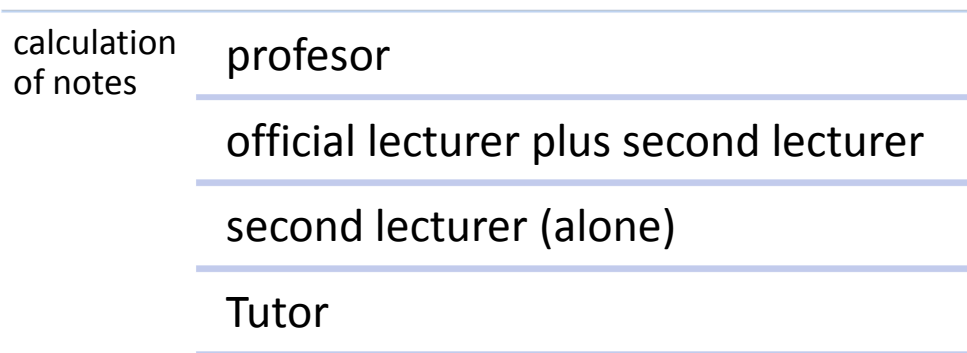
#### Adaptions which are made on the process:

The designed process was adapted from the exercise task. The points for the exercise sheets should be inserted one after another but this would require a loop because the exercise sheet is made specifically for the program Aristaflow. This program has the implemented rule that every designed graph must be well formed but the used tool ADONISCE2 doesn't have this rule. Therefore, the loop isn't simple to design so this requirement wasn't fulfilled. Next is the implementation of the maximum number of included exercise sheets which is fifteen. To make sure of this an or-gate would be used but because this is only a requirement which isn't

needed in the further process this requirement also wasn't designed. Lastly the grade is checked that if the grade is 1 then the note is directly 1 whereas in other cases the grade must be generated. This would cause a quicker way to go through the process but isn't really relevant because the calculation can be done even if the grade isn't a 1. The outcome will be the same but the whole process must be executed. So this was also not designed in the test process.

### 5.2.2 Department scenario

Contrary to the described structure of the first test case this example doesn't have such a hierarchy. As explained is the process for generating a grade. Normally the calculation of a school grade is simple: the relevant teacher has a grading system but in this case the affected institution is a university and so the grading process is a bit more complicated than in school (not only by law but also by the university structure). In this case the process can be performed by different people who can stand and process the sequence alone, the only reason for difficulties is the academic grade. According to the law [10] the process of grading is the responsibility of the teaching person but four different scenarios could be found to show how different people are combined to process the grades. The differences depend on how many people are involved with one course.



*Figure 12: This picture shows the 4 scenarios for calculating grades in a university course. These scenarios are based on personal experience.*

The first scenario is simple: a professor holds a course alone and so he is responsible for the grading. For the second case is set a professor with a second lecturer in case of absence. The third scenario includes a second lecturer holding the course alone but also the need for the presence of a professor as a main lecturer because of a lack of qualification. In this case the second lecturer only needs confirmation from the main lecturer for his calculated grades. The last identified scenario is the participation of a tutor. The usage of a tutor can lead to different extends. He can be used to hold a tutorial, takes part in the course as a contact person for questions from the students or in special cases even help grade the homework. In every case a tutor holds the responsibility of helping to convey the course material. In each of the four described cases the process of grading is the same for the most part but the number of involved people is different therefore the more people are involved the greater the process gets because communication and check activities are needed.



### 5.2.3 Individual views and the positions in the scenario

The source for this test case was a task from an exercise sheet therefore only one process exists. To get a whole scenario such as the one described for each existing person various views were generated based on personal experience in both process modelling and organisational situations therefore the following views came into existence. The whole process won't be explained but the original process model was described in a former part of this chapter. In the following during the introduction of the used views only the differences between the original process and the view will be explained. The description will be verbal and in form of the system from [9]. Furthermore, in the process view pictures the changes will be marked with colours. Green symbolises the addition of an activity, red symbolises the deletion of an activity existing in the already described original process and blue symbolises a position change of an activity. For the changed position the position where the activity, or more structural elements, will be sent to will be marked. The previous position can be found in the description.

#### View Professor:

Normally a professor is the responsible person for a course therefore this view will be used as the base for this test case because he holds the responsibility for the grading in every found scenario.

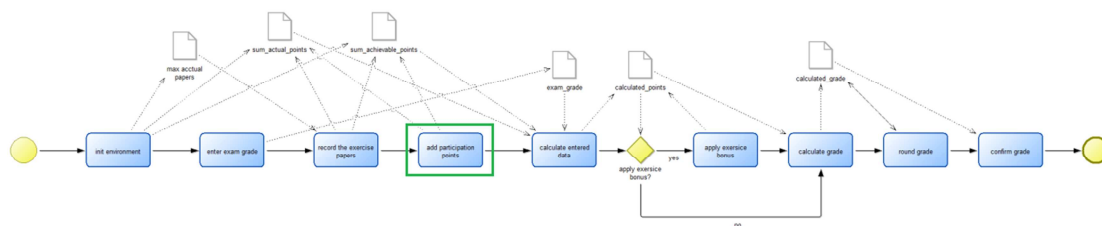


Figure 13: Shows the generated view for the professor which is used as the base view for the algorithm.

As can be seen in figure 13 the difference between the original process and the view of a professor only includes the addition of one activity. This activity is named “add participant points”. What does this activity include? The main idea of inserting this activity is based in the school system. In this system the behaviour of the students also takes part in the grading. Why is this not included in the original process? It is simple: The amount of students in one course can be much bigger as a school class. One course can be attended by hundred students or more and this amount is hard to handle with participant points. Why use it now? The number of students depends on the point advancement and the branch of study. According to the attractiveness of a branch more or fewer students participate but also the number of places in courses is limited. Nonetheless, the influence of the branch is high on the participation. To summarize: This activity was added to satisfy the scenario of small course groups where the professor has the chance to interact with his students.

*Activity: insert (professor, add participant points, record the exercise papers, calculate entered data)*



### View Official lecturer plus second lecturer:

This view shows the described situation that a professor and a second lecturer hold the course. The second lecturer can be a substitute when the professor can't hold the course or use it for training to hold courses alone.

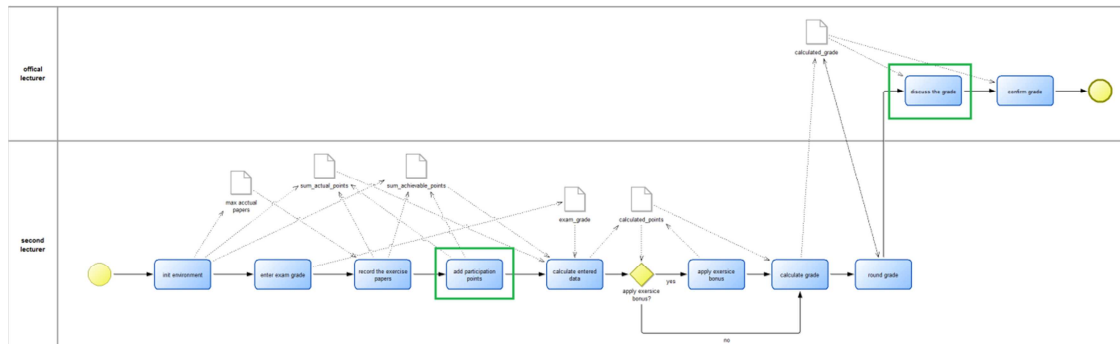


Figure 14: Shows the view which was generated based on the scenario that one lecturer can be substituted by another one because of absences.

The first inserted activity is the same that can be found in the professor's view: the usage of participant points. After that the second change is the addition of an activity “discuss the grade” which stands for the process that both lecturers of the course discuss the grades for each student. The impression of each student on both teachers can differ therefore this can have an effect on the grading between both parts which leads to a discussion.

Activity: *insert (official plus second lecturer, add participant points, record the exercise papers, calculate entered data)*

Activity: *insert (official plus second lecturer, discuss the grade, round grade, confirm grade)*

### View second lecturer:

This scenario takes into account that a lecturer alone holds the course but needs back up from a professor who is officially responsible for the course. The only parts in this scenario which are executed from the professor are the grade entering and the transferring of the grades.

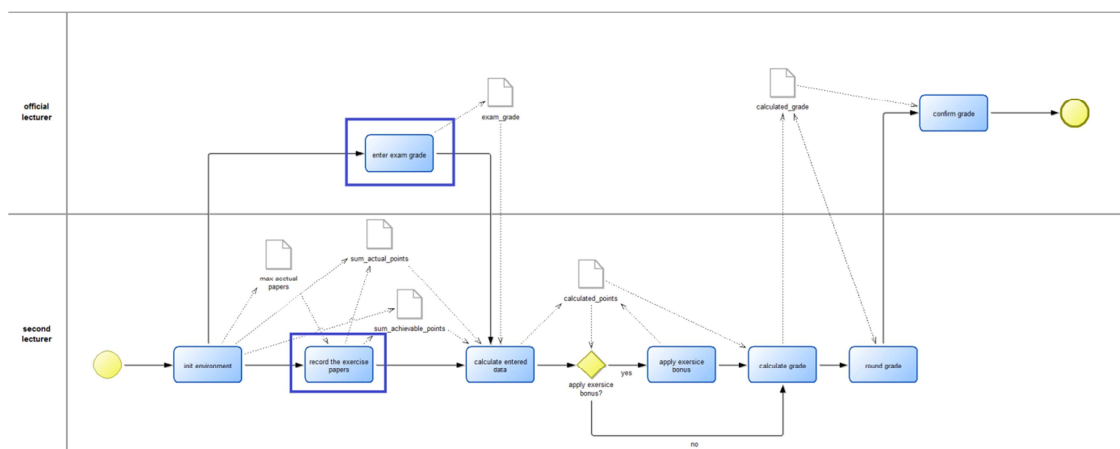


Figure 15: Shows the view of a lecture which has two lecturers but is managed by a single one and sometimes assisted by the superior.

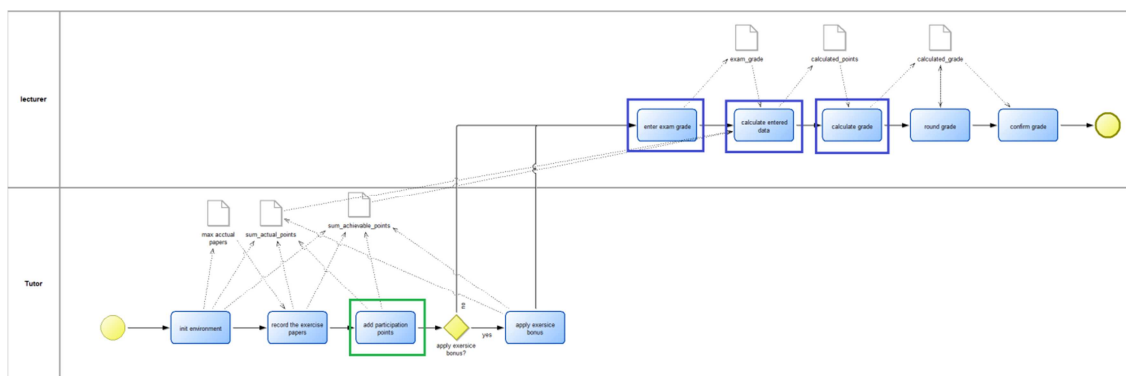
As explained the professor holds two responsibilities: The entering of the exam grade and the transferring of the notes. The sequence of the activities changed because the responsibility of grading lies by the professor alone. Now a parallel execution of homework and exam are performed by the second and official lecturer. This means only position changes appear in this view compared to the described original process.

*Activity: delete (second lecturer, enter exam grade)*

*Activity: pinsert (second lecturer, enter exam grade, init environment, calculate entered data)*

View Tutor:

At last the scenario of a tutor in this case is responsible for the homework, the correction and the grading of those. The exam and generation of a grade is not part of the tasks of the tutor as these tasks are executed by the professor.



*Figure 16: Shows a scenario packed into a view of a tutor who was used as a contact person for questions and also for checking the homework.*

Once more participant points are used and inserted which is part of the tutor's duty to evaluate. Due to the separation of exercise sheets and the exam a few activities are changed to a different position in the process. "Enter exam grade" and "calculate entered data" changed to a later position in the process after deciding if an exercise bonus is applied or not. The later activities from calculate grade changed their position too because of the former position change.

*Activity: delete (tutor, enter exam grade)*

*Activity: delete (tutor, calculate entered data)*

*Activity: delete (tutor, calculate grade)*

*Activity: sinsert (tutor, add participant points, record the exercise papers, apply exercise bonus?)*

*Activity: sinsert (tutor, enter exam grade, apply exercise bonus, round grade)*

*Activity: sinsert (tutor, calculate entered data, enter exam grade, round grade)*

*Activity: sinsert (tutor, calculate grade, calculate entered data, round grade)*

#### 5.2.4 Merged model

For this test example a limitation level structure and organisation level structure was generated by me because the test process is taken out of an exercise sheet so these information are not available in the task. Therefore, the scenarios were taken as a base to generate these two table contents (table 9.1 and 9.2) so not much more information is available. The limitation level was used with no restrictions but the organisation level 4 (see

table 9.2) is part of the structure. Nonetheless, no view is available therefore this level will not be used for the algorithm. The restricted area extends from level one to three.

limitation level	
1	tutor
2	Second lecturer
3	professor

organisation level	
1	tutor
2	second lecturer
3	professor
4	faculty chief

Table 9.1: Shows the limitation level structure which was developed based on the department scenario.

Table 9.2: This table shows the organisation level developed on the scenario of test process 2.

After showing the designed scenario for the limitation level (table 9.1) and organisation level (table 9.2) and before executing the merge, another table must be known, the final point table. This table is needed because without knowing the exact levels and counted numbers the algorithm doesn't work. In table 9.3 the values of the used factors can be seen for test example two: the grading process.

	# sub processes	Organisation level	# data connections	# connectors	# involved people	Limitation level
Professor	0	3	19	1	1	3
Official lecturer plus second lecturer	0	2	20	1	2	2
Second lecturer	0	2	17	1	2	2
tutor	0	1	19	1	2	1

Table 9.3: Shows the finished point table with all the gathered values for each view for test example 2.

Now all needed information for the developed algorithm is known so it can be executed on the views. In figure 17 the generated merge model can be seen. It's a small picture so a greater version can be found in the appendix 12.3. During the description of the views colour was used to better showcase the changes in the model. The same colouring was used in the merged model. Additionally, to the colour the description will also be in text form and after [9].

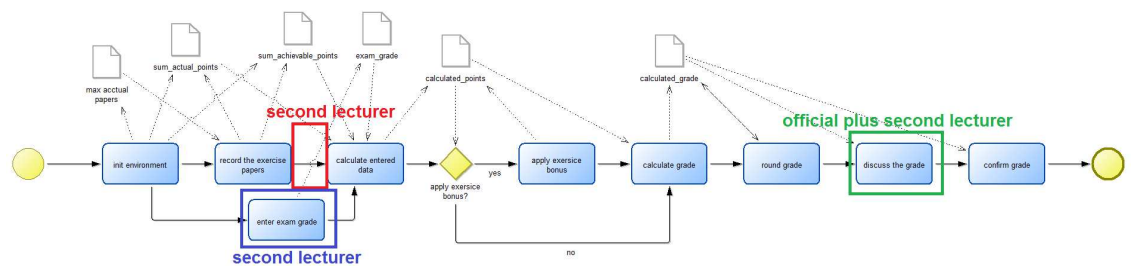


Figure 17: Shows the merged model which is generated over the developed algorithm. The colouring shows the changes.

The base process view is fixed as the view of the professor, so 3 other views must be merged into this base. The sequence of the views used in the algorithm is official lecturer plus second lecturer (short official plus second lecturer), second lecturer and tutor. So in the first run of the algorithm the base view professor and the second view official plus second lecturer are merged. Only one change could be found: the extra activity “discuss the grade” which is a communication activity between the two parts of the view official plus second lecturer. Therefore, this isn’t existing in the base because only one person is involved in this view so no communication to other parties exists. To summarize it up: this activity will be added to the base. In the next run the now adapted base view and the second lecturer view are merged. Two changes can be found: one deletion and one position change. The deletion is referred to the activity “add participant points”. The deletion is executed because the condition of the algorithm that the limitation level is maximum one level lower as the base is satisfied. Furthermore, the position change of the activity “enter exam grade” is taken out of the sequence and inserted into a parallel branch to the recording of the exam papers. The organisation level is appropriate for deletion and position change so the position change is also executed. In the last run the view tutor should be merged but because the limitation is the lowest possible level this view is excluded from the merge and no changes are adapted.

*Activity: delete (merge noting, add participant points)*

*Activity: delete (merge noting, enter exam grade)*

*Activity: pinsert (merge noting, enter exam grade, init environment, calculate entered data)*

*Activity: sinsert (merge noting, discuss the grade, round grade, confirm grade)*

## 5.3 Testing case KFZ

In this chapter the last test case for KFZ insurance will be described. Before the testing is executed the process itself is explained. Only one process is known because this test case is based on a task of a university course named Workflow Technologies from the University of Vienna. This process is set as the original process and the other views are generated out of this original view. After explaining the original process the test scenario of the working department is explained but because this information isn’t available in the task the scenario was constructed on personal experience. The corresponding views of the scenario are explained after the scenario itself. At last the generated merge from the test is explained.

### 5.3.1 Introduction of the process

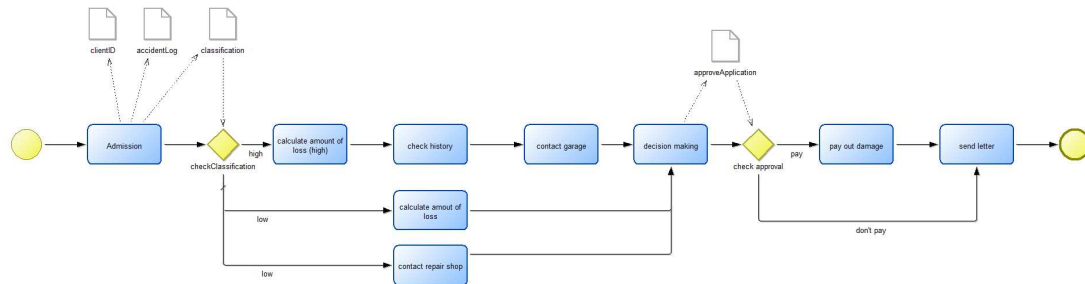
So which process is taken in this test? It says KFZ but what it really means is an insured event, the process of the insurance and what they do when such a case for a KFZ arrives. There are quite big differences in the cases where it all depends on the circumstances - of the accident case and the history of the customer - if an insurance will be paid or not by the company. The process itself is complex even in a simple case so the following used task from an exercise sheet of the course Workflow Technologies (WS 13/14) will only be a further abstraction of a real process. The used task is as follows:

*„Jede Schadensmeldung eines Versicherten wird von einem Mitarbeiter der Schadensabteilung registriert. Danach wird sie von einem Sachverständigen nach gering oder gross klassifiziert. Für geringe Schäden wird die Versicherungssumme ermittelt und parallel dazu die Garage kontaktiert. Bei grossen Fällen werden - in dieser Reihenfolge - die Versicherungssumme ermittelt, die Schadenshistorie ermittelt und die Garage kontaktiert. Für beide Schadensarten wird danach eine Entscheidung getroffen. Es gibt zwei mögliche Ergebnisse: OK oder nicht OK.“*

*Ist das Ergebnis OK, zahlt die Versicherung. Ein Mitarbeiter der Finanzabteilung veranlasst die Auszahlung. Ist das Ergebnis nicht OK, wird nicht gezahlt. In beiden Fällen, wird ein Brief an den Versicherten geschickt, der ihm die Entscheidung mitgeteilt.“*

*Source: Rinderle-Ma S., Course Workflow Technologies, exercise sheet 2 exercise 2.2 from winter semester 2013/2014 University of Vienna*

The following process was designed after this used task. The picture is small but a greater version of it can be found in the appendix 12.4.



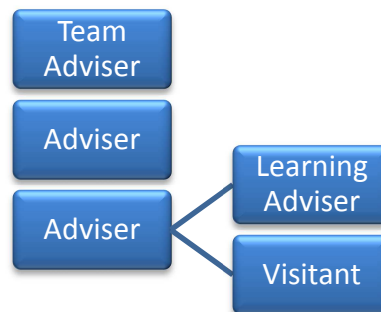
*Figure 18: Shows the designed process model based on a task of an exercise sheet from the course Workflow Technologies of the University of Vienna which is used as original process.*

First of all a start event is needed, after that is an activity which is there to signal that three pieces of information must be known for the further process. These are the following: who is the relevant client, the accident log and a classification of the incident. Now the classification will be executed through an or-gate. If the classification shows that the incident will cost as much as the category high then the first path will be executed. In the sequence the amount of potential loss will be executed at first after that the history of the client will be examined and lastly the garage will be contacted to find out what the damage is. If the calculated category is low then in a parallel execution of path two and three the garage will be contacted and the amount of damage will be calculated. One path will be chosen which represents the process of making the decision to pay for the damage or not. After making the decision the process will be evaluated through an or-gate. If the insurance company will pay then the activity pay out the damage will be executed. Whether the company pays or not the client receives a letter with the decision and an explanation in every case which leads to the end of the process.

### 5.3.2 Department scenario

In this special case two different approaches of the described process are followed. This process includes an advising activity which isn't modelled and can be executed in different forms. How it's all done depends on the company politics. One approach shows the regular policy that one customer has a specific adviser. It is the same system as an adviser in a bank. The customer opens an account by an adviser which from then on is his adviser until the point he changes companies, position in the company or the customer switches to another company. This system also takes place in other branches and influenced this test process. The other approach is based on the explained case which has one big problem. The customer is too dependent of his assigned adviser. If for example this adviser is currently on holiday urgent business is proceeding slower because another adviser must take over and analyse the current situation of the customer. Only after an investigation the adviser can do any kind of guidance. To avoid this scenario another strategy for guiding the customers was created. Now a customer doesn't have one specific adviser but a whole adviser team. The team is based on a branch and every employee has to be included in the decisions for the customers. This has the

benefit that the customer isn't dependent on one adviser and the decisions aren't made by one person. One disadvantage of this approach can be that the decision making can take some time because more people mean more communication. So these two approaches will be represented by 3 views including 2 advisers and 1 example for the team approach.



*Figure 19: This picture shows two different approaches for customer support plus two more constructs of occurring people in this scenario.*

Why is the hierarchy of the scenario (figure 19) only two levels and which concepts are used to design it? In a branch many people are working and they have one branch leader well as a substitute leader. In this case, contrary to test process 1 a deep hierarchy is not possible because each adviser must have the same rights regardless of their position in the branch hierarchy. The position couldn't be included in this scenario because the rights are the same mostly the same due to the same responsibilities. Nevertheless, the case that a new employee comes and another adviser must educate him can also be performed with this structure. After all this means that no learning adviser can make a deal alone until he matures to a fully educated adviser but the right level of both - the learning and teaching advisor - are the same. The last implemented situation is the company concept of a visitor which was taken for the last position. This concept is based on the idea that employees of other departments change their work place for the period of one week for further education. With this, every employee can study the other departments what brings a better understanding for the work of other departments but also for the whole company. For example, I myself experienced this system because in a bank the rule is that every new employee must complete a week in a branch to learn how the daily work with the customers is done. This is naturally only for those people who are not employed to work in a branch. These four concepts were taken to develop the scenario for this test process.

### 5.3.3 Individual views and the positions in the scenario

The scenarios for this test case were all made out of the constructed scenarios in the insurance branch, just as in the grading test case. These already explained scenarios were taken and views were generated. The source for the adaptations of the views is based on the scenario but they are implemented through my personal experience for processes. Now the different generated views will be shortly explained in text form but also in a more formal description after [9]. Not the whole process of the views will be described but only the difference between the defined original process to the view. Furthermore, because the pictures of the views are small they can be found bigger in the appendix 12.4. To show the differences in the graph in a better way without explanation the changes were highlighted with colour. Green stands for insertion, red for deletion and blue for a position change of an activity.

### Adviser:

First of all, the view which will be used as a base is the view of an adviser. This adviser is a full-fledged employee who can handle a client without further intervention.

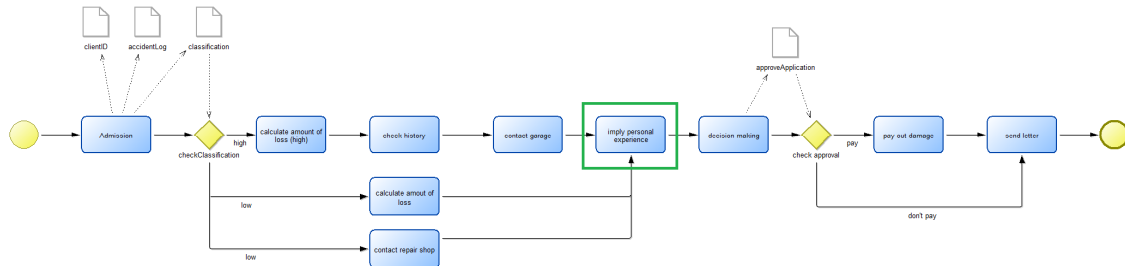


Figure 20: Shows the base process from the test process KFZ.

The only thing that changed between the described original process and this view is that additionally to the reports now the personal opinion of the adviser is included. After checking the reports and before making the decision a new activity named “imply personal experience” takes place which holds the process of weighting between the decision and the personal experience with the client. Such a factor isn’t unusual in this branch because a model alone can’t show if the person has potential or not. A person can decide if the future additional benefits could be greater when a specific decision will be made now. To take this in account this view was generated.

Activity: sInsert (Adviser, imply personal experience, contact garage, decision making)

### Adviser:

Now another adviser will be explained. This view (figure 21) is based on the scenario that the adviser of a client is on holiday and another adviser takes the job but must get approval for the final decision from the original adviser.

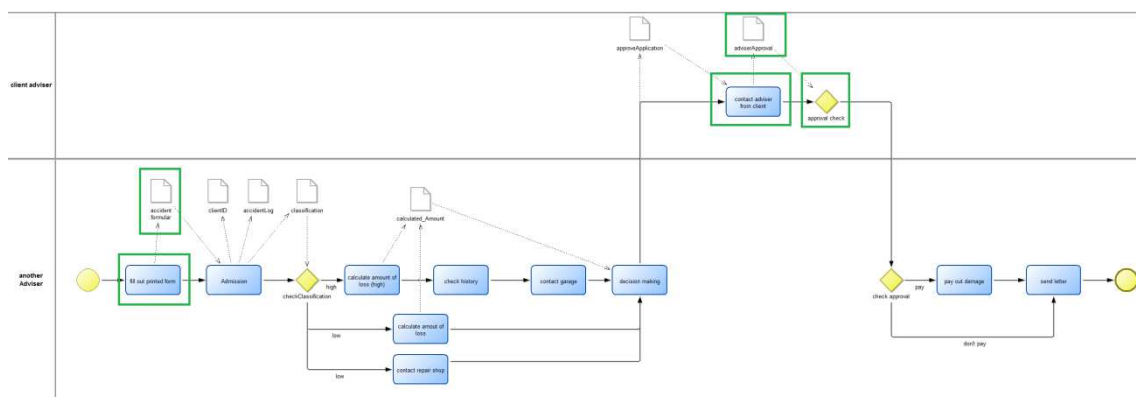


Figure 21: Shows the constructed view for an adviser which is based on the case that the adviser substitutes for another adviser who is on holiday. So the whole process could be executed by him but a last approval from the actual adviser is needed.

What changes when two advisers take on the job? Firstly, the representative adviser must fill out a printed form for the protocol after which the process goes forth to the decision making. After making the decision the approval of the real adviser of this client must be got and when



this is successfully solved the last evaluation can proceed. To summarize: two activities, two data objects and one examination which tests if the approval is got are added.

*Activity: sInsert (Adviser, fill out printed form, start event, Admission)*

*Data Object: Insert (Adviser, accident formular, fill out printed form, Admission)*

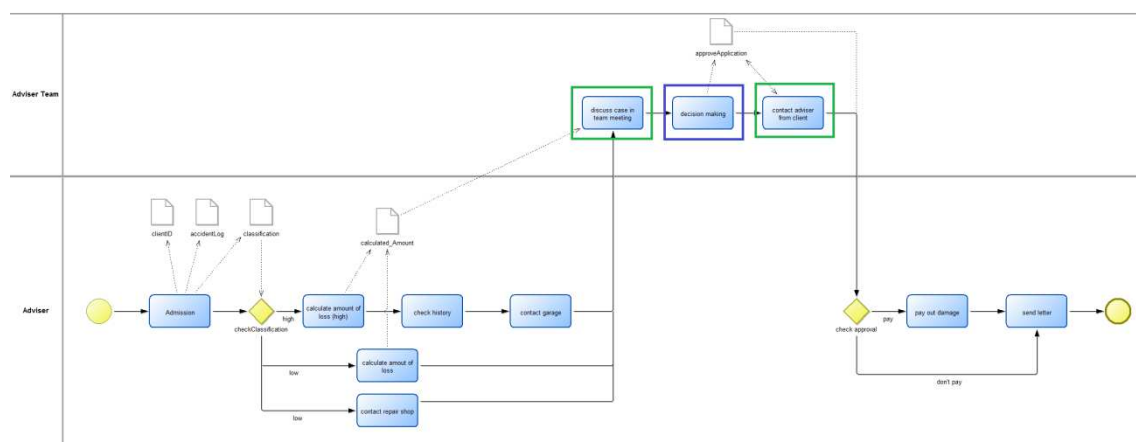
*Activity: sInsert (Adviser, contact adviser from client, decision making, check approval?)*

*Connector: sInsert (Adviser, approval check?, contact adviser from client, check approval)*

*Data Object: Insert (Adviser, adviserApproval, contact adviser from client, approval check?)*

### Team Adviser:

This situation was already explained but to summarize it: each client doesn't have a specific adviser but rather a whole team. The advising is realised by one person but the decisions are made from all people in the team.



*Figure 22: Shows the designed view for the position team adviser of the scenario. The designed main idea is that no decision can be made alone but only in a team.*

What changes in comparison to the original process? The advising is made by one person and only the part of where the decision is made changes. This includes that a new activity named "discuss case in team meeting" is implemented before the actual act of decision making is executed. Every case must be discussed in the team because no specific adviser is fixed for a client therefore more people can have an impression on one client. After discussion a decision must be made so the already existing activity changed its position to this point. The position change only appears in the logic, in form of inserting the objects the activity is on the same place but because the predecessor and ancestor are changed the activity was marked as position change. Lastly a new activity where the adviser of the client receives the decision is implemented. It is meant that the same adviser which executed the procedure of advising transfers the decision to the client. This is set because it is better that one case is entirely processed by the same adviser.

*Activity: sInsert (team adviser, discuss case in team meeting, contact garage, decision making)*

*Activity: sInsert (team adviser, contact adviser from client, decision making, check approval)*



### Learning Adviser:

The job of a learning adviser is to learn the processes and how to give advice to a client. Therefore, it is fastest to learn by observing the real process executed from another adviser. It is easier to gather the needed knowledge by observing because no responsibilities lie on the learning adviser.

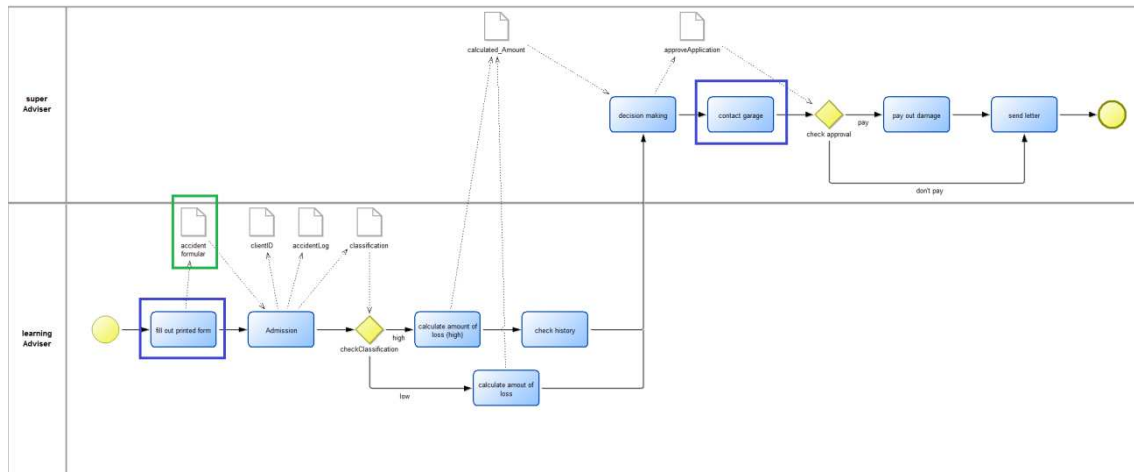


Figure 23: Shows the view of the learning adviser who has all rights and can perform the process alone but needs guidance to avoid failures.

For learning and also documenting the process an activity fill out a printed form is needed therefore this activity is added at the beginning of the process. Furthermore, a responsible task is to get the report of the garage so this task is part of the work of the instructing adviser and not of the learning adviser. Therefore, this activity changes position to a later point in the process and the decision making takes place before the report is obtained. The reason behind this change is a small adaptation which includes not using the garage report for the decision making.

Activity: *insert* (learning adviser, fill out printed form, start event, Admission)

Activity: *delete* (learning adviser, contact garage)

Activity: *delete* (learning adviser, contact repair shop)

Activity: *insert* (learning adviser, contact garage, decision making, check approval)

### Visitor:

This case brings out the situation of a person from another department in the same company or even the same department who wants to learn another business process used in the company on his own. Therefore, no actual part of the process is performed by this "visitor" but the whole process will be overlooked by him.

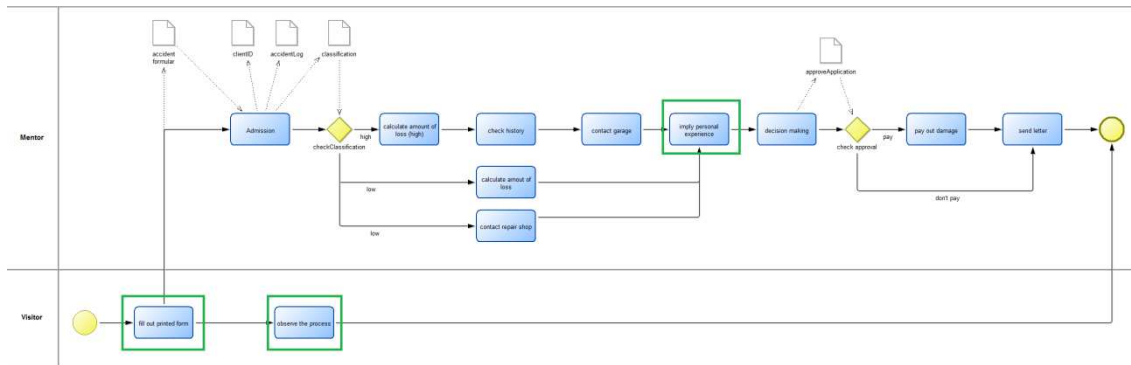


Figure 24: Shows a specially constructed system used in real life. The scenario was taken and adapted on this department's scenario.

The only thing changing during the real process is implementing the personal experience with a client for the decision making. This change also occurs in other views therefore this is an already explained thematic change. The left two insertions of activities come from the visitor who holds the observing action. The processes are “filling out a printed form” and “observing the process”. Why is the already explained process of filling out the printed form done by the visitor? This only requires filling out an existing form it can be done by the visitor by following instructions of the mentoring adviser. So the writing is executed by the visitor but the content of the filled out form is supported by the adviser.

Activity: *sInsert (visitor, fill out printed form, start event, Admission)*

Data Object: *Insert (visitor, accident formular, fill out printed form, Admission)*

Activity: *plnsert(visitor, observe the process, fill out printed form, end event)*

Activity: *sInsert (visitor, imply personal experience, contact garage, decision making)*

### 5.3.4 Merged model

Like test case 2, this test case is also based on a task of an exercise sheet from the university course Workflow Technologies so the level structure is entirely made by me. It's also based on the scenario structure which was generated. The organisation structure is easier to construct therefore in comparison to the limitation level the organisation structure is restricted. The area goes from level 1 to 5 so only level 6 isn't counted because no view for this level 6 exists in the generated scenario.

limitation level	
0	visitor
1	learning adviser
2	adviser limited
3	adviser / team

Table 10.1: This table shows the developed limitation levels which were constructed out of personal experience and the test scenario.

organisation level	
1	visitor
2	learning adviser
3	adviser limited
4	adviser
5	team
6	department chief

Table 10.2: Shows the related organisation levels to the limitation levels. The process of generation is based on the same sources.

Now the level structure will be exercised on the organisation structure and the cooperating views to get the level for each view. The newly produced information in combination with the

other factors, form the point table. After gathering every needed information the point table for this test case 3 is completed and can be seen in table 10.3.

	# sub processes	Organisation level	# data connections	# connectors	# involved people	Limitation level
<b>Adviser</b>	0	4	6	2	1	3
<b>Adviser</b>	0	4	9	2	2	3
<b>Team adviser</b>	0	3	13	3	2	2
<b>Learning adviser</b>	0	2	11	2	2	1
<b>visitor</b>	0	1	8	2	2	0

Table 10.3: Shows the point table with inserted values for the factors of the developed algorithm. The values are based on the organisation and limitation level from table 10.1 and 10.2 and on the views where the remaining factors were counted out.

After explaining the relevant tables, the generated merge will be described. The merged model can be seen in figure 25. The picture is so small that it is hard to read it so a bigger version can be found in the appendix 12.4. As can be seen the changes are marked with the same colouring which was used for the view differences. In addition to the colour the origin of the change is written above the object. In the following description the outcome of the used algorithm will be explained verbally and over statements after [9].

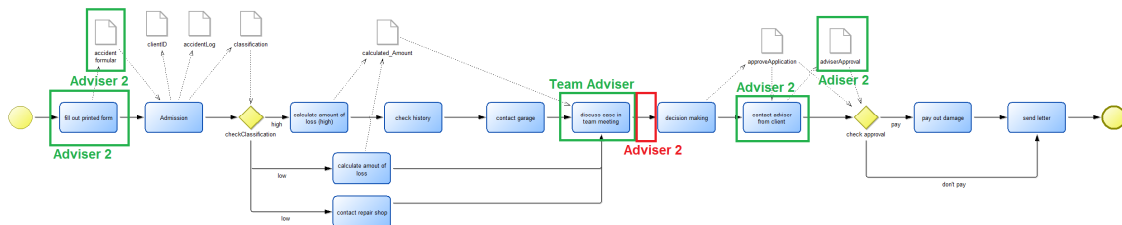


Figure 25: Shows the merge model for the process KFZ which was generated using a base view and using the algorithm to merge the other views into the base view.

The merged model is constructed out of all scenario views but the algorithm can only handle one view at a time therefore each view was added after a specific order. The used order is adviser, team adviser, learning adviser and visitor. A base view must be defined and the best views to be used like that are the two advisor views because in the scenario two advisers appear. Hence, the base view will now be adviser 1 and the other adviser is adviser 2. In the first run of the algorithm the base view and the view of adviser 2 are merged. The changes which occur are 2 new activities and the deletion of one existing activity. First the first activity of the process is changed because a new activity called “fill out printed form” is inserted. Second, the deletion is the change of the activity “imply personal experience” because it doesn’t exist in the view of adviser 2 and the organisation level of the view of adviser 2 is at maximum one step lower than the base view level the deletion is executed. The last change which comes from this view is the insertion of the activity “contact adviser from client” plus

the corresponding data object “adviserApproval”. These new activity and data object are both structural elements which are needed because in the view of adviser 2 a communication between two people exists so it is now merged into the base process. The next view which is used for the merge is the view of the team adviser instead of a single adviser. Two changes are added from this view and both are insertions. First, a new activity named “discuss case in team meeting” is inserted before the decision making. The second change from this view isn’t implemented because it is already inserted in the merge with the view of adviser 2 therefore this change already exists in the base view. The next run-through is merging the current base view with the view of the learning adviser. Also two changes occur but both are not executed because one change is the insertion of an activity which is already inserted by a former view and the second one is a position change of the view “contact garage”. This position change is not executed because of the algorithm where it is defined that only position changes from views with an organisation level which are at maximum one level lower are taken. This view has a difference of two levels therefore this change isn’t implemented. The last run-through is used for the view of the visitor which isn’t implemented in the base model because the limitation level is the lowest one which is by definition in the algorithm excluded from the merge.

*Activity: sInsert (merge KFZ, fill out printed form, start event, Admission)*

*Data Object: Insert (merge KFZ, accident formular, fill out printed form, Admission)*

*Activity: replace (merge KFZ, imply personal experience, discuss case in team meeting)*

*Activity: sInsert (merge KFZ, contact Adviser from client, decision making, check approval)*

*Data Object: Insert (merge KFZ, adviserApproval, contact Adviser from client, check approval)*

## 5.4 Summary of the testing

In this chapter the testing of the developed algorithm was executed. For the testing three different processes from different branches were taken. The first one is a process named “Prolima control” of the company Raiffeisenlandesbank Niederösterreich-Wien. At the time of making this thesis I was employed in the mentioned company and the process was provided to me for this master thesis with the agreement of the management. The other two used processes are based on tasks from an exercise sheet of the course Workflow Technologies from the University of Vienna. This course was part of my studies so the exercise sheet was available. All three processes are explained in the same structure which is the following: first introduction of the process itself, after that the department scenario is explained followed by the corresponding views. Lastly, the generated merged model is described. The first process was taken out of real life therefore the department structure is also the real life structure and the views were designed over interviews with the people of the company. The other two processes based on the sheet only deliver the original process so the department structure and views were designed by me. In addition to the merge description the used point tables, the limitation level as well as the organisation level structure are shown and described. The views and the level structures come from real life in the first example and the other two examples were completely self-designed.

## 6 Evaluation of the testing

At this point the merged models for all test processes are generated. The interesting question at this point is how close the generated model is to the original model. To check this, evaluations are made in this chapter to show it. For each test case an evaluation is executed to find out how many percent of the original model can be found in the merged model. Because this is a rather vague number a second evaluation is made where not the percentage of equality but the structural elements are essential. This means that in the second evaluation the merged model is analysed on how many elements are new, deleted, changed their position or are simply unchanged. The purpose of these evaluations is to find out how close the generated model of the developed approach can be to the original process.

### 6.1 Evaluation on equality of activities

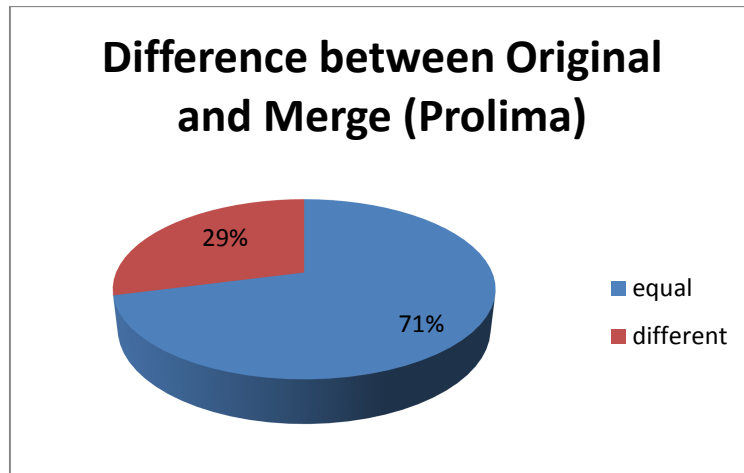
In this chapter the first evaluation is described. The purpose of this evaluation is to find out how many structural elements of the original process can be found in the generated merged model. For this purpose each of the used three test processes will be executed separately. The procedure of figuring this out was simple. The sequence of the structural elements from the original process and the merged model were listed in an excel sheet. With structural elements the existing activities and data objects are included while the links between activities and data objects are excluded. It was done this way because in the description of the views or the merge this also wasn't included and the operations show high level change operations and not low level primitives based on [9]. The defined rule that the same activity name refers to the same element is also the reason a comparison of existing activities in both lists can be made. The logic used in excel was the following:

$$= WENN (original\ element = merge\ element ; 1 ; 0)$$

The idea is that the number 1 represents equality and 0 does not. To get a correct result it was necessary that elements which are only in one list have an empty field in the place of the element in the other list. When no empty spaces are in the list at the first difference the rest of the list is entirely wrong. After developing so far the differences are known but the percentage isn't. To get the percentage the maximum number of structural elements from both lists is taken. Afterwards one hundred is divided by this maximum number to get the percentage of one structural element or each list. Now two other lists are made with the following logic:

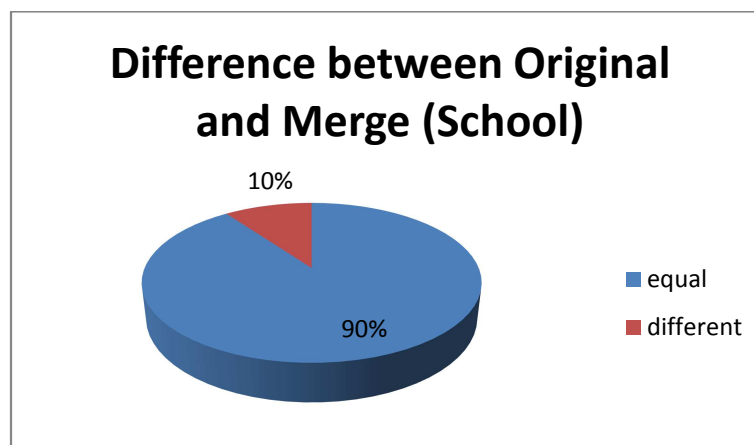
$$= WENN(equal = 1; percent\ number\ for\ one\ element; 0)$$

"Equal" in this logic means the outcome from the logic before is either zero or one. It is a simple logic which takes the calculated percent number for one element if the equality is 1. This is made for both lists to generate two new lists with the calculated percentages. When these two lists are taken to create a graph such as a pie chart the merged model list is set to one hundred percent. So when the original model has less percent in every case then it shows how much percent of the original model are in the merged model. Now the three results of the evaluation are shown.



*Figure 26: Shows the evaluation of how much of the original process is in the generated merged model for the test example Prolima control.*

This pie chart (figure 26) shows the result of the described evaluation. What can be seen is that 71 percent of the merged model also appears in the original model. 29 percent are not equal to a structural element in the merged model. What can't be seen in this evaluation is that each structural element of the original model also appears in the merged model and that the remaining 29 percent are new elements. These elements were taken out of the views which were different from each other but the important parts of the process which appear in the original are all present.



*Figure 27: Shows the difference between the original model to the merged model in percent. The process is the grading process from a university course.*

This pie chart (figure 27) refers to the second test case process of the grading for a university course. This result is better than the one from the Prolima control process because it shows that 90 percent of the original process can be found in the merged model. As already explained in the description of the model from the Prolima process this model only shows how much percent of the merged model are also in the original model. What it doesn't show is that the 10 percent are not in the original model because the structural element for this part are from the views and they only appear in the merged model. Furthermore, the difference between appearance and absence in both models is so small because the process scenario was

the smallest out of the three. This shows that the number of variations is also a factor for the merge because a lower number brings the merged model closer to the original model.

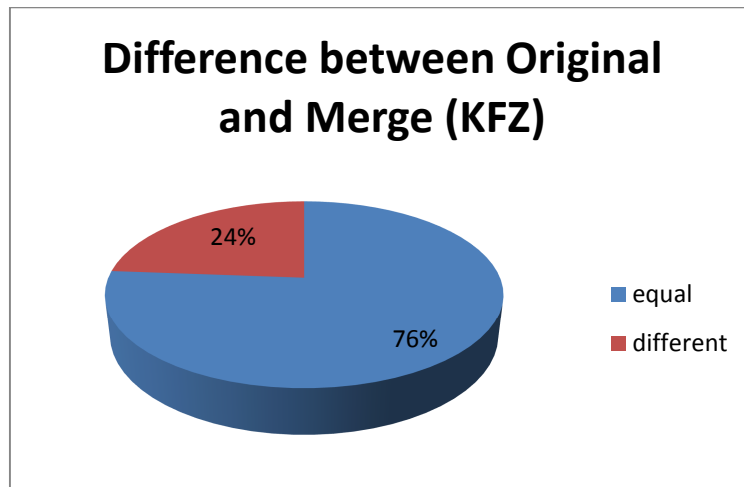


Figure 28: Shows the structural elements in percent and how much of the elements are equal between the original model and the merged model.

Lastly, the evaluation model from the KFZ insurance process (figure 28) will be analysed. The result of this evaluation is close to the result from the first evaluation of the Prolima control process. 76 percent of the original model can be found in the merged model. As in the other models it isn't evident that the whole original model appears in the merged model and that the difference comes from additional structural elements. Because this model is closer to the first evaluation it can be assumed that the second evaluation is an exception because of too little variety in the number of views.

## 6.2 Evaluation of the structural differences

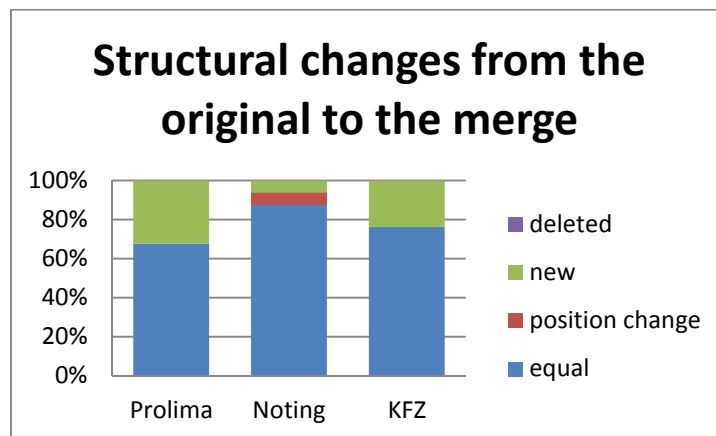
To overcome the gap of the former evaluation where the result is rather misleading another evaluation is made to show the structural changes between the original models to the merged models. For this purpose a table (table 11) was created where the structure is noted. For each test process case the number of equal, new, deleted and activities with a changed position can be seen.

	Prolima	University noting	KFZ insurance
equal	21	14	16
position change	0	1	0
new	10	1	5
deleted	0	0	0

Table 11: Shows the generated table with the values for the structural changes from the original model to the merged model. For all three test cases these values were calculated.

The gathering of this information was simple. The first list was taken from the last evaluation where the process structure of both models is defined. Out of this list each information can be got by simple counting. No logic in excel or any another kind of algorithm was used. The reason behind the counting is the time factor. It was the easiest and fastest way because a list with all the elements already existed.

After defining table 11 with the counted values for each structural change, a bar chart for all three test cases was designed. In this chart (figure 29) the vertical line shows a total of one hundred percent which refers to the merged model structure. The base value for each tower is the total number of structural elements from the merged models.



*Figure 29: Shows the structural changes from the original to the merged model. Equal and position change shows that these activities are in both models. New means an extra activity in the merged model which isn't in the original. The deletion part is important because it represents missing elements from the original model in the merged model.*

What can be seen in the three cases is that the number of equal activities is over 60 percent. As known from the former evaluation it's even more than 70 percent but this was already known so what exactly can be seen in this evaluation? The important parts are the deletions because a deletion implies that an activity from the original process doesn't exist in the merged model. As can be seen in the model not one activity of the original processes is missing. Furthermore, position changes are shown because it signifies that an activity is in both models but on different positions. At this point a business modeller must take the merged model and decide if adaptations are needed or not. To summarise it: all relevant structural elements from the original models can be found in the merged model.

### 6.3 Summary of the Evaluations

In this chapter the generated merged models from the developed algorithm were evaluated. Two different questions were answered over two evaluations:

1. How close is the merged model to the original model?
2. How many structural changes exist between the original model and the merged model?

To answer the first question: all three test processes achieved an equality of over 70 percent. The remaining 30 percent include structural elements which aren't in the original model. This means that the merged model is greater than the original. Therefore, 100 percent equality can only be reached when the original model is the outcome of the logic. So without the second evaluation the result of the first evaluation can't be interpreted correctly because it can't be seen if the whole original process is in the merged model. In the second evaluation the structural changes are shown. The main reason for this evaluation is to show if deletions are executed which would mean that a part of the original model isn't in the merged model. In



these three test cases no deletion was performed. In the noting process one position change occurred but in total every generated model contains the whole original process.

## 7 Discussion

In this master thesis the main focus was set on the development of a logic for the defined model which is described in chapter research method. In the development of the logic the usage of a system to figure out if an activity of the view is the same as the one in another view was essential. In the literature [11] [12] [7] [13] different methods for the purpose of finding similarity were found. Why wasn't one of those used? The use of such an algorithm or logic would demand an analysis of the different methods where the best one can be selected on set requirements of which outcome should be accomplished. For this purpose each found logic must be tested on more than one example because the executed tests in this paper are from different branches so a solution for only one branch can't be used. This would require a lot of examples, time and a bigger analysis not only on the correctness of the outcome but also on the flexibility with graphs of varied size. The flexibility is needed because the main idea of the developed algorithm is to deal with any kind of model. So in the end it was decided that the benefit of such an analysis is too low at this point of developing the approach.

Another part of the work was to use scenarios of departments or another form of a hierarchical structure which can be used to get information such as an organisation level. The limitation level can even exist when no hierarchy exists but the organisation level is important in the developed approach. In fact only one real life scenario could be received the other two used examples were made by me. Why did I develop the used scenarios all by myself and did not seek help from other experts or looked for real life scenarios? To use a real life scenario the approval of someone is needed, may it be a company, a branch or a person. Therefore, the search is based on the fact that the approval is given. Asking other experts for help brings the organisational responsibility. Even if the hurdles of finding people, time, a place and document the meeting can be overcome the problem of getting a good output for scenarios cannot be guaranteed. This is because human's opinions differ so a clear model that satisfies for example three people is difficult to make. Furthermore, the self-developed scenarios are based on real life situations I experienced myself therefore the scenario is set up but the individual views are based on real life concepts. It must be noted that these scenarios are taken from different branches but the different situations didn't need adaptations to fit in the branch of the scenario. In the end the generated scenarios show real life character which was the reason why I designed the test scenario departments by myself.

An important part of this paper is that the process and scenario was defined and after that the views were generated. The first test case process was the only one where information was gathered by interviewing the people of a real department but after the interviews are done the process must be analysed and the different views must be generated. This step was completely left out in this work. Why wasn't this included in the work? The reason behind it is that no source for all three test processes exists therefore the use of one for only a single process would be too inefficient. For using such a source first it must exist and after that a method must be chosen but this can only be decided over an analysis. The different approaches use different sources, therefore a source must be found and the permission to use the source must be given. To avoid all these difficulties for this part and because the focus of

the master thesis lies on developing the explained approach the part of generating the views for the work wasn't included.

Another point that wasn't included in this paper is the benefit of using this approach. What is the benefit of using this method compared to others? If the processes are first modelled or adapted because of a change in the company the information must be gathered with any kind of method. For example when using questionnaires for this purpose it can be modified to fit this approach. By adapting the method of gaining information the process can be automated so that in a further step views are generated and when views exist in a form where the structure is saved this method can be used. Of the other approaches found in the literature (can be found in chapter 9) most provide only a method for the information gathering and defining the views. This approach starts at the point where the others end. Therefore, the benefit lies in the fact that a part of the workload from a business modeller can be lifted. This approach cannot handle the whole work of a modeller but it can provide a model which can be adapted to fit in the final requirements. So a part of the work which consumes a big chunk of time can be completed by a machine.

Why wasn't the logic programmed and executed? In this master thesis it was attempted to find another way to approach a merge between different business process views into one model. Because the development of this approach included the use of a definite tool and needed a similarity check the programming would be rather difficult. First if it was programmed it would be in the tool's internal language which could only be useful to people who use this specific tool. Second as already explained a further analysis would be needed to find a suitable method of finding similarity between the activities. Before even thinking about starting the programming act these hurdles must be taken therefore the amount of time to even get to the point of taking action in programming was classified as too high.

Lastly the testing of this approach was made for the purpose to show that the method can be used on processes from different branches. It is simple to show that it is no solution for a special case compared to evaluations about the behaviour of the logic. What wasn't evaluated is what effect it has on the logic if the used process is bigger and more complex. Another point is that the number of variations doesn't differ too much among the three test cases. A deeper analysis in this direction would be good. If the behaviour of the logic in cases with bigger processes or more variations is known a better analysis can be made to decide if the logic is better than others or not.

## **8 Future work**

Based on the described open points from chapter 7 this developed method has great potential. First of all the algorithm should be programmed and tested if any adaptations must be made because some cases can occur in the process of programming that weren't obvious during the process of writing the pseudo code. Then a further evaluation with a greater amount of different processes should be made to explain the behaviour of the algorithm. After that it should be tested in a real life situation where the merged model can be analysed with the help of a business modeller and the benefit can be set. Before this developed approach can be called a true and useful method these steps must be executed and the result should be in a defined area which was declared in advance. The result of this master thesis is only a

foundation stone. Without further work no building can be built and only the fundament which is described in this master thesis is left. The last found point is that the algorithm must be adapted that the goal of difference the changes in one view in some to implement into the merge and to ignore the rest. In the work it was already tried to implement this function but at the point of the evaluation it was clear that the generated logic can't differentiate between changes to take for the merge and others not to take out of the same view. This could be possible by using for views with more than one involved person the tasks which are executed from another person are treated with the levels from this other person. The described scenario was the only way I could think of to achieve this so this adaptation should be tried out and implemented but the testing shouldn't be forgotten.

## 9 Related work

As starting point for the literature research the following literature [3] [5] [7] [14] was used for the snowball method. Keywords and additional literature was extracted from the starting literature. In a second step an internet research, with the found keywords, was performed (merging processes, business process, view, merging, merging business process views, metamodel and merging algorithm). Most found information for merging was from the field software engineering because the topic of merging is well discussed in that field. In the field of modelling and business processes, merging is still rather new therefore only a view approaches exist. These approaches differ in modelling languages and the set goal for merging. In the end the references were limited to a part of the found literature. The reference list at the end of this thesis will only show the for this master thesis inspiring works. The related work covers a number of different sectors therefore the fields of work differ. In this chapter the references are presented in order of general information for modelling and merging. After that the main part of this work introduces the different merge approaches. The state-of-the-art methods differ in actual merging of some kind of model and taking part in the cycle of first implementing a model system. The second part covers how to design the system in a specific way to let the processes be recorded in a specific way so that a merge isn't needed. Now the different references will be explained in the described order.

It was essential for this master thesis that different forms of information are known such as the modelling. The modelling of the views comes with the problem that errors in the models can bring a false output. Therefore, [2] shows the most potential failures that every modeller should know because the responsibility of modelling lies by the business modeller. However, no explicit or definite way to create a model exists. Modelling is an individual process so the same model would look different when using various tools because the modelling languages differ in the structural elements. So by using different languages the same model can't come out the same. Another point of this master thesis was how to get correct information for the different business views. In [3] it is described how important the right questioning is to find enough information for the models. In a continuation paper [15] can be seen that by using a 3D model for simulating the working process information can be got which would not be possible by only using a questionnaire. So the conclusion of these two references is that the way how modelling information is gathered is an important part. Another question was how to perform the merge in this master thesis. An overview over the existing different merge algorithms from the software engineering field can be found in [5] where each approach is

shortly explained together with the corresponding problems. The connection between the software engineering field to the modelling side in case of refactoring can be found in [16]. Back to [5] after describing the different methods a difference algorithm is presented which purpose it is to find differences between two models.

At last a great part of merging is to find equality or similarities in the different models. In this field a large amount of different approaches can be found between simply “checking the labels of two activities” to “form a corresponding map” where correlations between the activities are shown [11] [12]. Other approaches work with the behaviour of the process itself as model; [7] shows taking the behaviour combined with the execution log [11]. In [13] a method is shown where the labelling of the activities is taken and formatted to eliminate plural forms of the words and delete all epithets such as “the” or “of”. The purpose is to get the labels as close as possible to find similarities by checking the edited labels.

[17] presents a special form of using a 3-way merge. The main idea is that all relevant models are based on the same parent model. With this in mind the procedure has 3 steps. First to find out what changes exist in the three models, what is inserted or deleted, etcetera. In the second step the merged model will be generated by taking into account that collisions occur. For these cases step three comes in where a modeller takes the merged model and adapts it until the collisions are solved and the merge is successfully evaluated. Another method for merging models can be found in [18] where the main idea is that the merging process is easier by using directed graphs. The merging algorithm must satisfy three defined criteria: first both inputs must be reflected in the output. Second, each element has to be retraceable to the source in the corresponding model and the last criterion is that out of the merged model both source models can be once more generated. An example with two models is executed and analysed. Other types of merging also occur where the models aren’t directly merged. Nevertheless, there’s another form of information from where a model can be generated out of. In [19] [20] [14] different methods for merging, event driven process chains (EPCs) are explained. In [3] another method is introduced by using the merge logic on log information which are first transformed and then merged. A model based on the merged information is created.

Other approaches like implementing a framework in the life cycle [21] or using another method are also forms to get the models into a form where no merging is needed because no duplicate exists [22] [16].

## 10 Summary

The goal of this master thesis was to find a way for merging different business process views of the same process. After developing the first idea of a research method a strategy was created on how the implementation of the idea should work. The strategy was then designed as a graphic model. A big part of the work went into finding the proper factors which would allow such a merge. These factors symbolise information in a business process model. At first a list was generated with potential factors that were taken out of the structure of a model and also from the used tool ADONISCE2. After deciding which criteria these factors must satisfy the found potential factors were analysed and then reduced to the essential ones. After the analysis a logic was developed which executed the merge. In this thesis the individual steps of the creation of the algorithm are described. To test the new approach three different case study processes were taken and developed into a department scenario to simulate the real life situation of a number of variations for the same process. The examples are different to each other because they come from different branches. To summarize it: the first test process is taken from the finance sector and was provided by a real company. The other two examples handle a grading process for a university course and a car insurance case. The source of the content was an exercise sheet of the university course “Workflow Technologies” from the University of Vienna. For these two examples the simulations of the departments are self-made but real life situations where used as pattern. After executing the developed algorithm on the views from the situations an evaluation of the results was made. The results have been analysed twice to see how close the generated model is to the original. Both examinations served the purpose to find out if the whole original model is shown in the merged model. This test was executed for all three test cases. In the last step the open points were discussed and summarized in the chapter 8 “Future work”.

## 11 References

- [1] Scholz-Reiter B., Business process modelling, New York, NY: Springer, 1996.
- [2] Rosemann M., „Potential pitfalls of process modeling: part A,“ *Business Process Management Journal* 12(2); pp. 249-254, 2006.
- [3] K. S. Rinderle-Ma S., „Human-centered process engineering based on content analysis and process view aggregation,“ in *Advanced Information Systems Engineering, volume 6741 of Lecture Notes in Computer Science*, Springer Berlin Heidelberg, 2011, pp. 467-481.
- [4] G.-B. L. D. R. M. Dumas M., „Similarity Search of Business Process Models,“ *IEEE Data Eng. Bull.* 32. Jg., Nr. 3, pp. 23-28, 2009.
- [5] Mens T., „A state-of-the-art survey on software merging,“ *Software Engineering, IEEE Transactions on*, 28(5):449-462, May 2002.
- [6] „BOC-Group,“ [Online]. Available: <http://www.boc-group.com/at/>. [Zugriff am 02 04 2015].
- [7] K. R. M. J. D. R. D. M. van Dongen B., „Similarity of business process models: Metrics and evaluation,“ in *Information Systems, 36(2): Special Issue: Semantic Integration of Data, Multimedia, and Services*, 2011, pp. 489-516.
- [8] D. M. O. C. Dijkman R. M., „Semantics and analysis of business process models in BPMN,“ *Information and Software Technology, Volume 50, Issue 12*, pp. 1281-1294, November 2008.
- [9] K. U. R. S. R. M. Jurisch M., „On representing purging, and utilizing change logs in process management systems,“ in *Business Process Management, volume 4102 of Lecture Notes in Computer Science*, Springer Berlin Heidelberg, 2006, pp. 241-256.
- [10] „Satzung Studienrecht § 8 Abs. (1),“ 01 03 2015. [Online]. Available: <http://satzung.univie.ac.at/studienrecht/>. [Zugriff am 04 04 2015].
- [11] B. M. Laue R., „A comparative survey of business process similarity measures,“ *Computers in Industry*, 63(2):148-167, 2012.
- [12] R. D. L. R. M. D. M. Uba R., „Business process model merging: An approach to business process consolidation,“ *ACM trans. Soft. Eng. Methodol.* 22(2):11:1-11:42, March 2013.
- [13] S. R. N. M. S. M. Schulte S., „Comparison and retrieval of process models using related cluster pairs,“ *Computers in Industry*, 63(2):168-180, 2012.
- [14] R. M. van der Aalst W.M.P., „A configurable reference modelling language,“ *Information Systems*, 32(1):1 - 23, 2007.

- [15] K.-F. S. B. R. R.-M. S. Krigelstein S., „Augmenting and assisting model elicitation tasks with 3d virtual world context metadata,” in *On the Move to Meaningful Internet Systems: OTM 2014 Conferences, volume 8841 of Lecture Notes in Computer Science*, Springer Berlin Heidelberg, 2014, pp. 39-56.
- [16] R. H. W. B. R. M. Mendling J., „Refactoring large process model repositories,” *Computers in Industry*, 62(5):467 - 486, 2011.
- [17] R.-M. S. Böhmer K., „Difference-preserving process merge,” in *On the Move to Meaningful Internet Systems: volume 8186 of Lecture Notes in Computer Science*, Springer Berlin Heidelberg, 2013, pp. 718 - 721.
- [18] D. R. L. R. M. D. M. Uba R., „Merging business process models,” in *On the Move to Meaningful Internet Systems: OTM 2010, volume 6426 of Lecture Notes in Computer Science, pages 96-113*, Springer Berlin Heidelberg, 2010.
- [19] G. F. v. d. A. W. Jansen-Vullers M., „Merging event-driven process chains,” in *On the Move to Meaningful Internet Systems: OTM 2008, volume 5331 of Lecture Notes in Computer Science*, Springer Berlin Heidelberg, 2008, pp. 418-426.
- [20] M. J. L. R. M. D. M. H.M. ter Hofstede A., „Configurable multiperspective business process models,” *Information Systems*, 36(2):313 - 340, 2011. Special Issue: Semantic Integration of Data, Multimedia, and Services, 2011.
- [21] H. A. B. T. Reichert M., „Capturing variability in business process models: the provop approach,” *Journal of Software Maintenance and Evolution: Research and Practice*, 22(6-7):519-546, 2010.
- [22] R. H. M. R. van der Toorn R.A., „Improved model management with aggregated business process models,” *Data Knowledge Engineering*, 68(2):221 -243, 2009.

## 12 Appendix

### 12.1 Interview guideline (German)

#### Fragen zum Ergebnis des Prozesses:

- Was ist das Ergebnis des Prozesses?
- Kann das Ergebnis unterschiedlich ausfallen?
- Wie groß können die Unterschiede sein?
- Anzahl der gefundenen Probleme?
- Doppelte Einträge?

#### Fragen zum Ablauf:

- Was ist der erste Arbeitsschritt?
- Gibt es eine Reihenfolge in der die einzelnen Schritte ausgeführt werden?
  - Kann diese Reihenfolge umgestellt werden?
  - Führt dies trotz allem zum selben Ergebnis?
- Welche Programme werden verwendet?
- Gibt es Daten oder Dateien auf die zugegriffen wird?
- Gibt es etwas das nur der Computer macht?
- Gibt es Dinge die per Hand nachgeschaut werden müssen?
  - Gibt es Einschränkungen durch Berechtigungen für Programme?
- Wie oft wird der Prozess von Ihnen in einer Woche ca. ausgeführt?

#### Fragen zu Kollegen:

- Wird der Prozess alleine bearbeitet?
  - Wenn nein, welcher Kollege wird zu Hilfe gerufen?
  - Wenn diese Person nicht da ist, wer wird dann befragt?
- Wird ein Kollege mit mehr Rechten auf Programme um Hilfe gebeten, oder wird der weitere Prozess einem Kollegen mit mehr Rechten übergeben?
  - Wird der Abschluss des Prozesses alleine ausgeführt, oder mit einem Kollegen?



## 12.2 Prolima control process

### Original

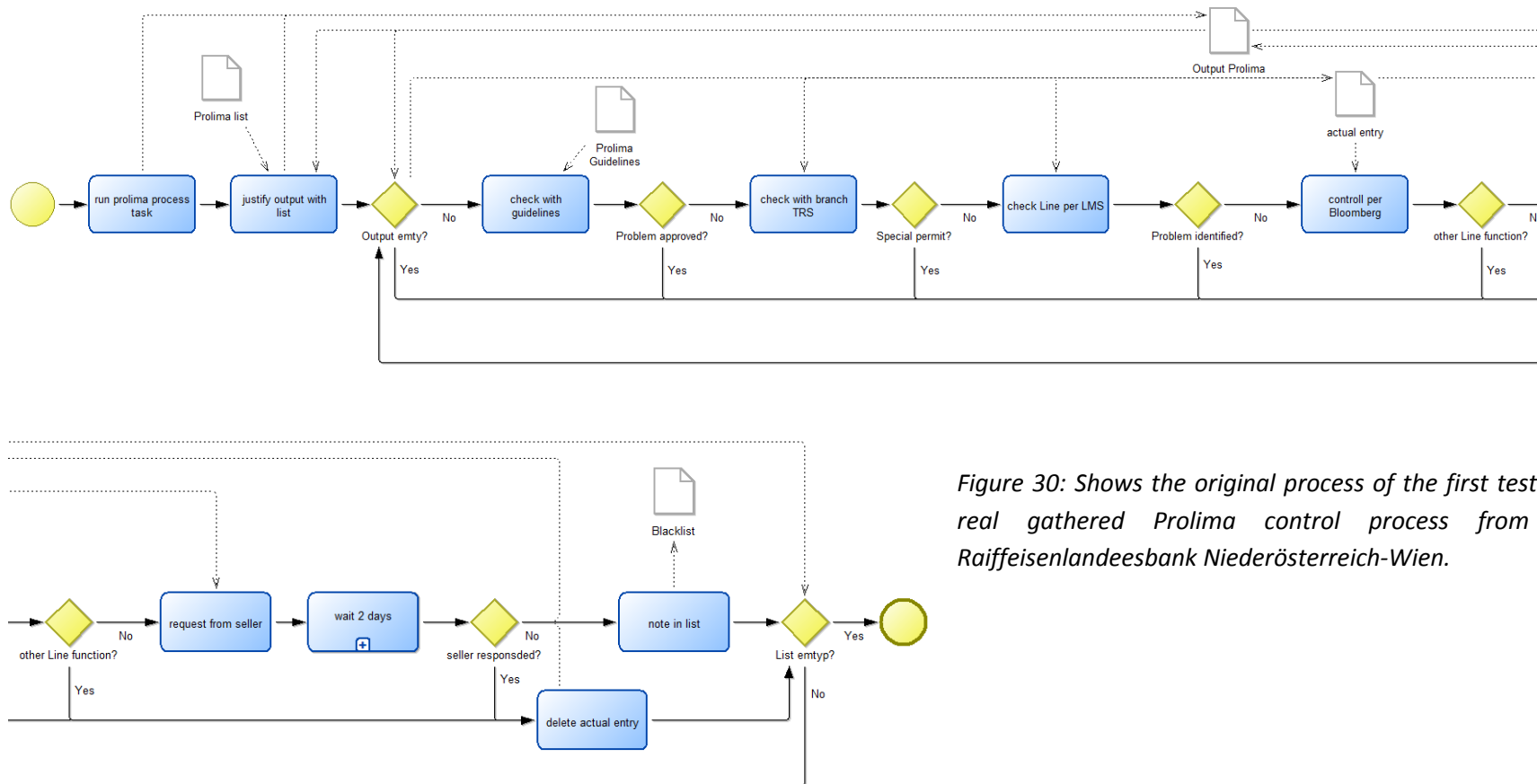


Figure 30: Shows the original process of the first test case. This is the real gathered Prolima control process from the company Raiffeisenlandesbank Niederösterreich-Wien.

## Base: team leader

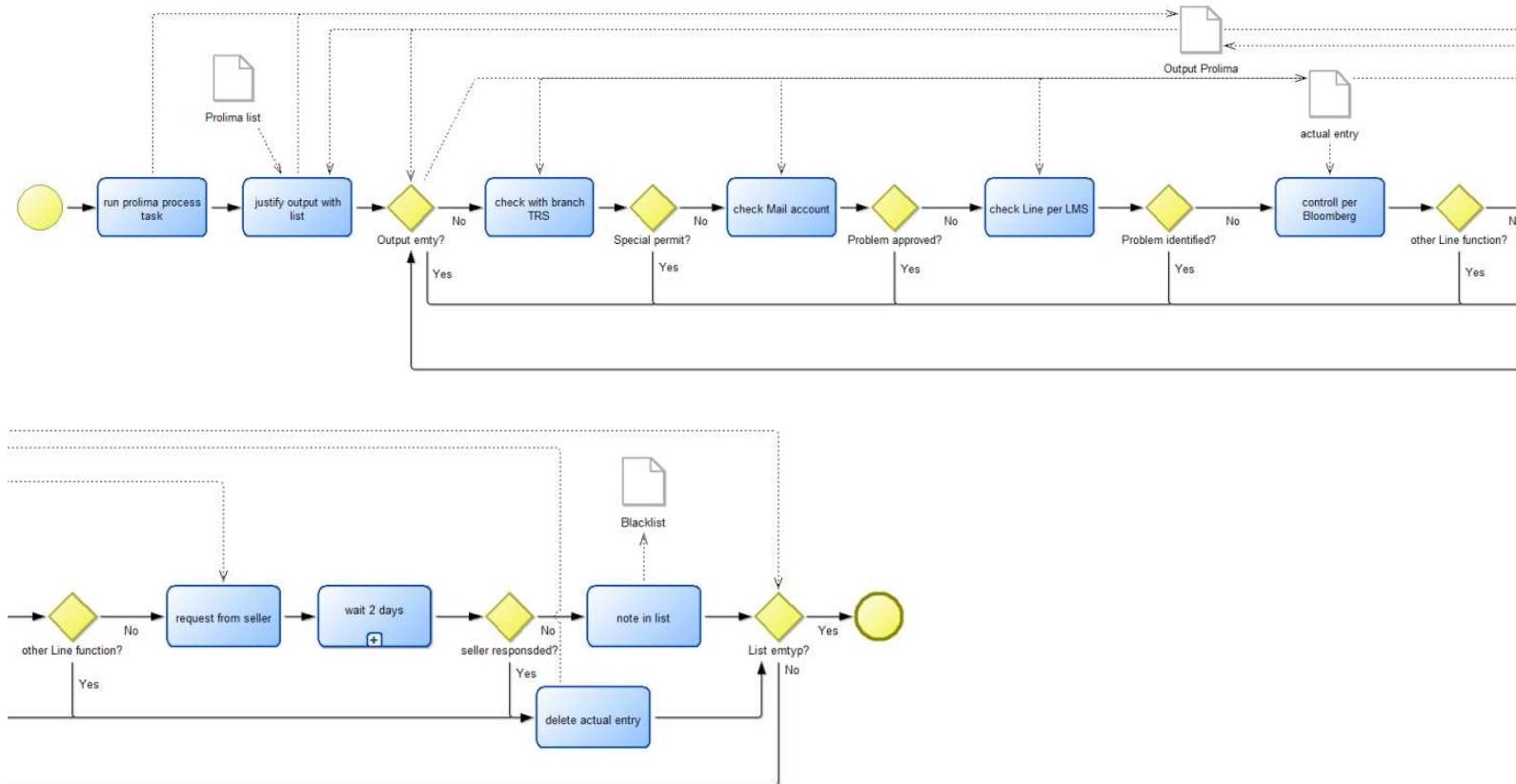


Figure 31: Shows the used base process for the first test process Prolima control. The person who owns this view is the team leader of the group Market Risk Management.

Base: team leader colour

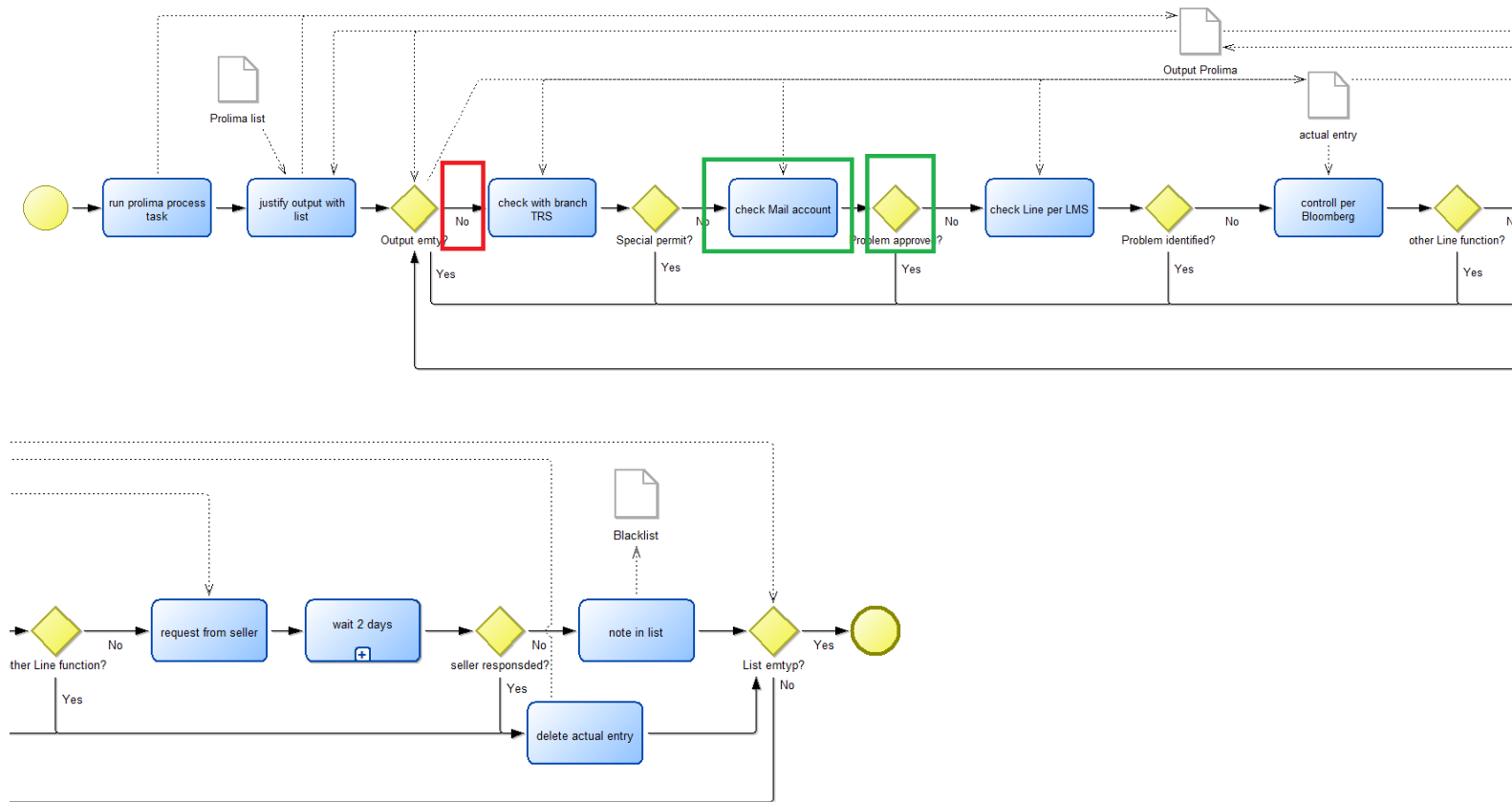


Figure 32: Shows the same picture as number 31. The difference shows through the highlighted changes from the original process (picture 30) to this view.

## Employee 1

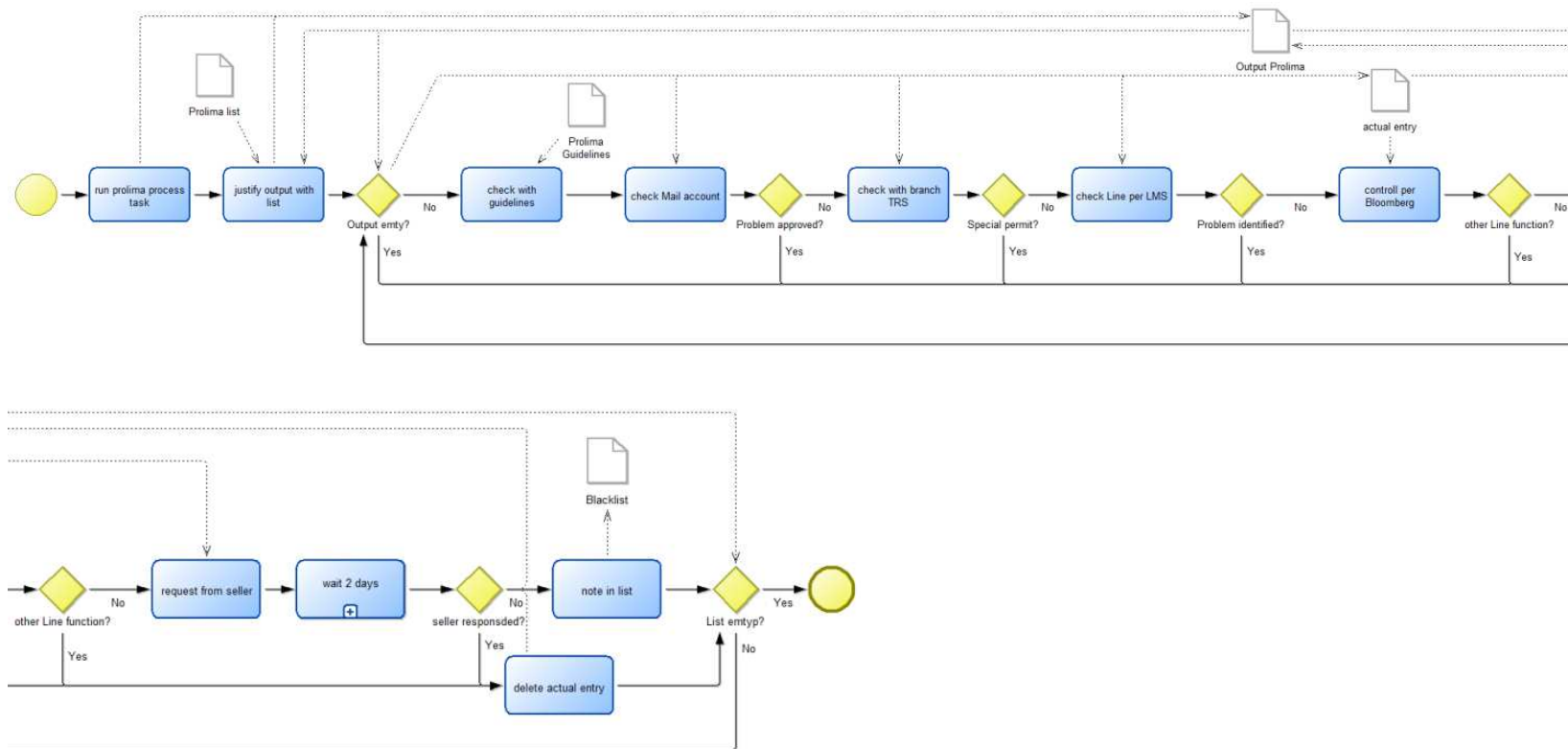


Figure 33: Shows a view from the test scenario of test process one. The owner of this view is named employee 1.

## Employee 1 colour

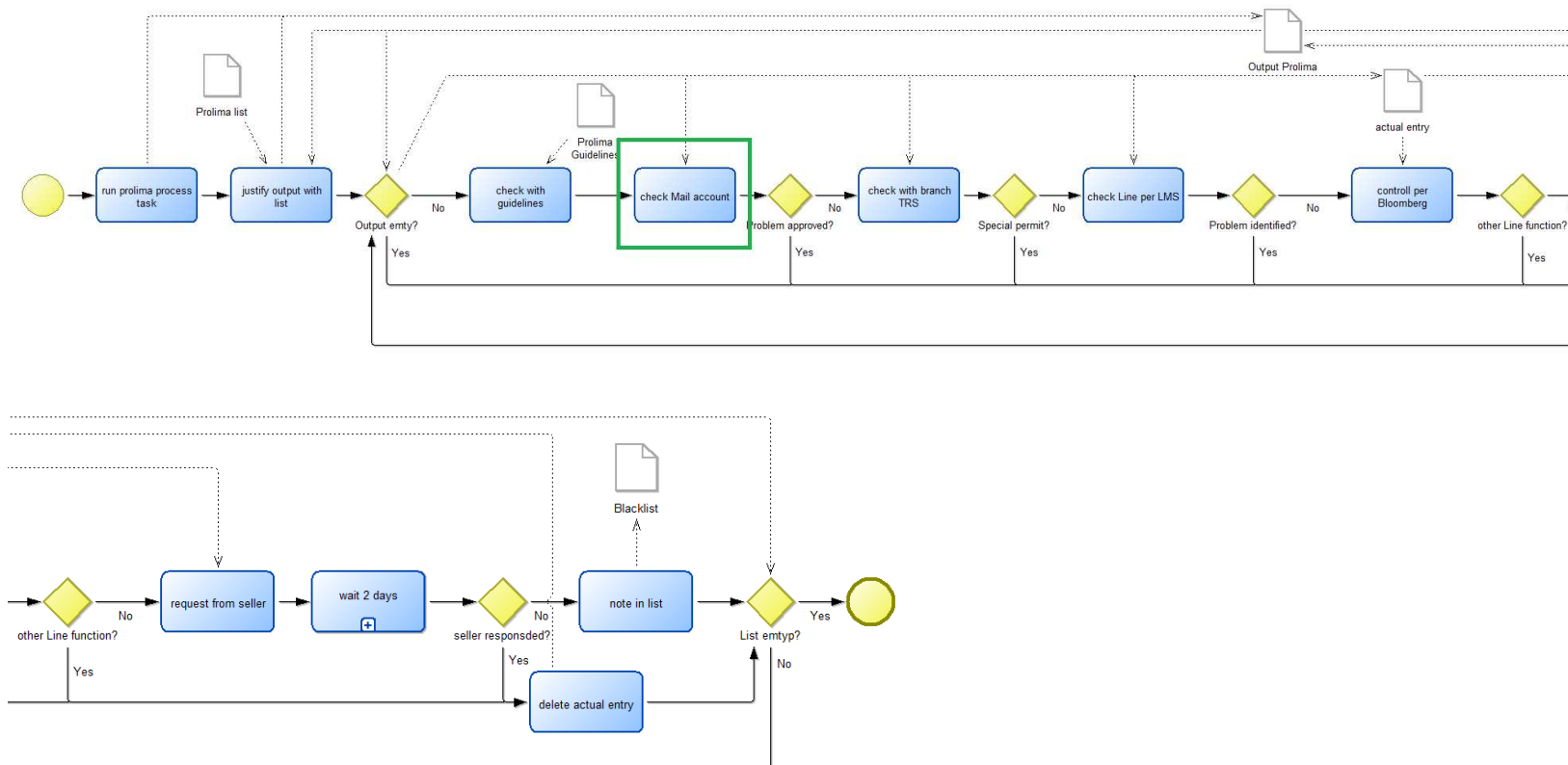


Figure 34: Shows the changes from the employee 1 (picture 33) to the original process (picture 30). Only one change appears; it's an insertion of an activity.

## Employee 2

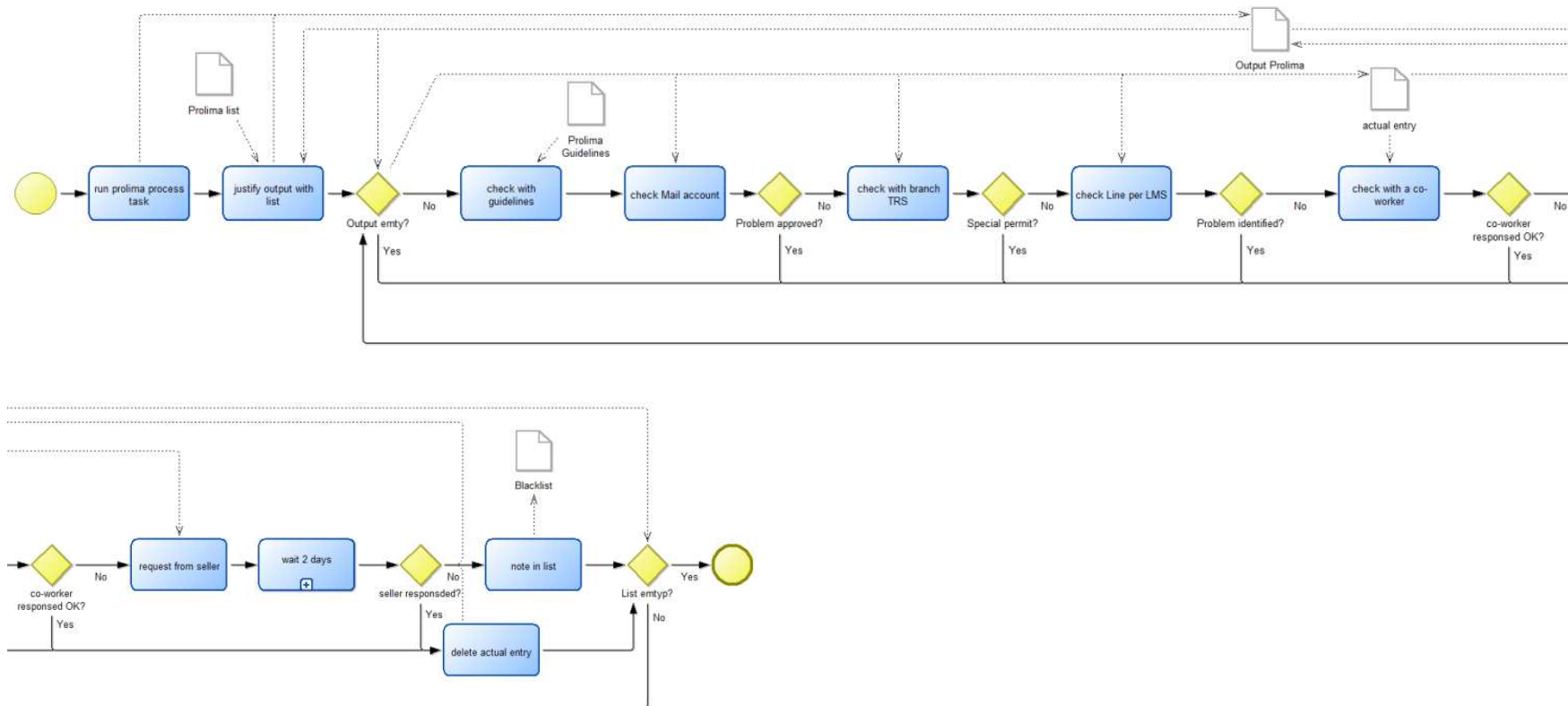


Figure 35: Shows another view of the first test process named employee 2.

## Employee 2 colour

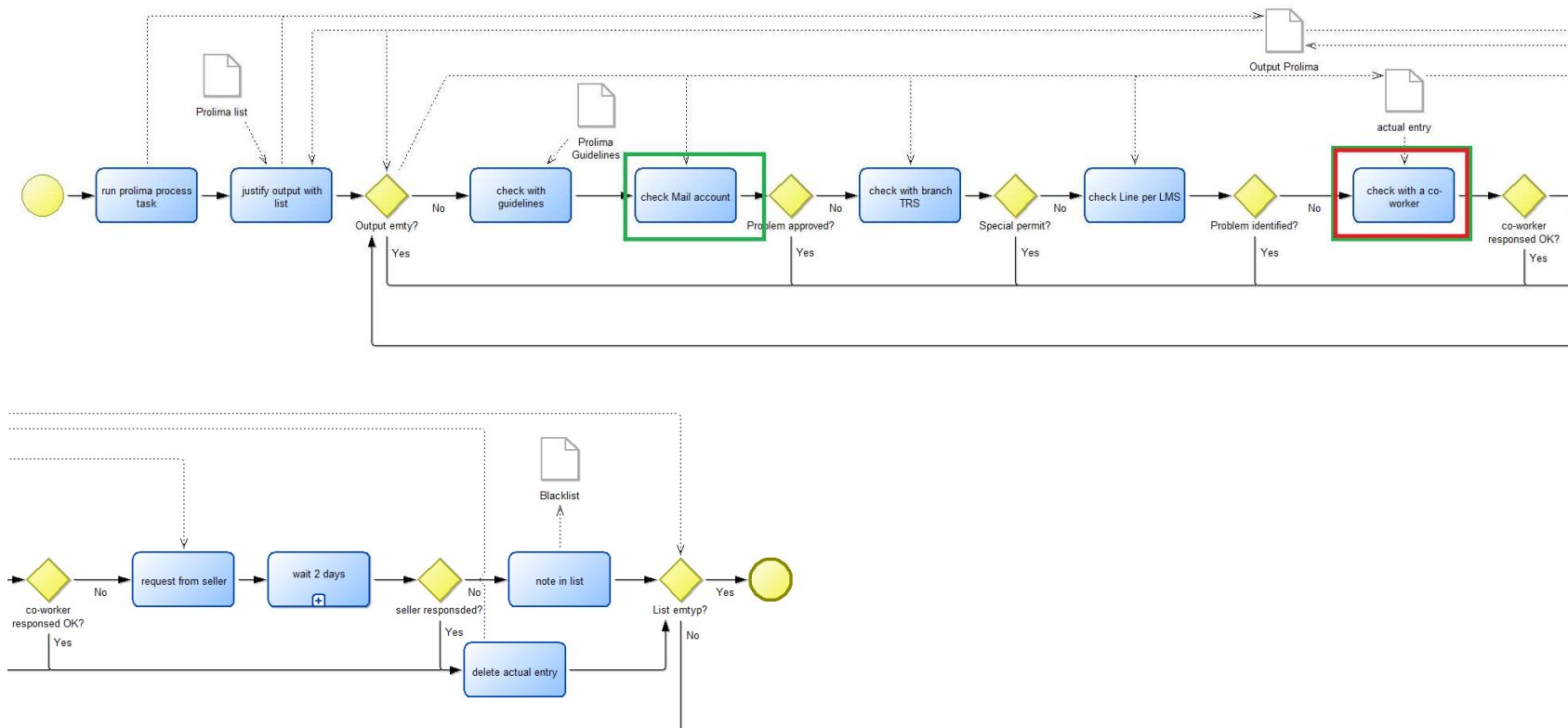
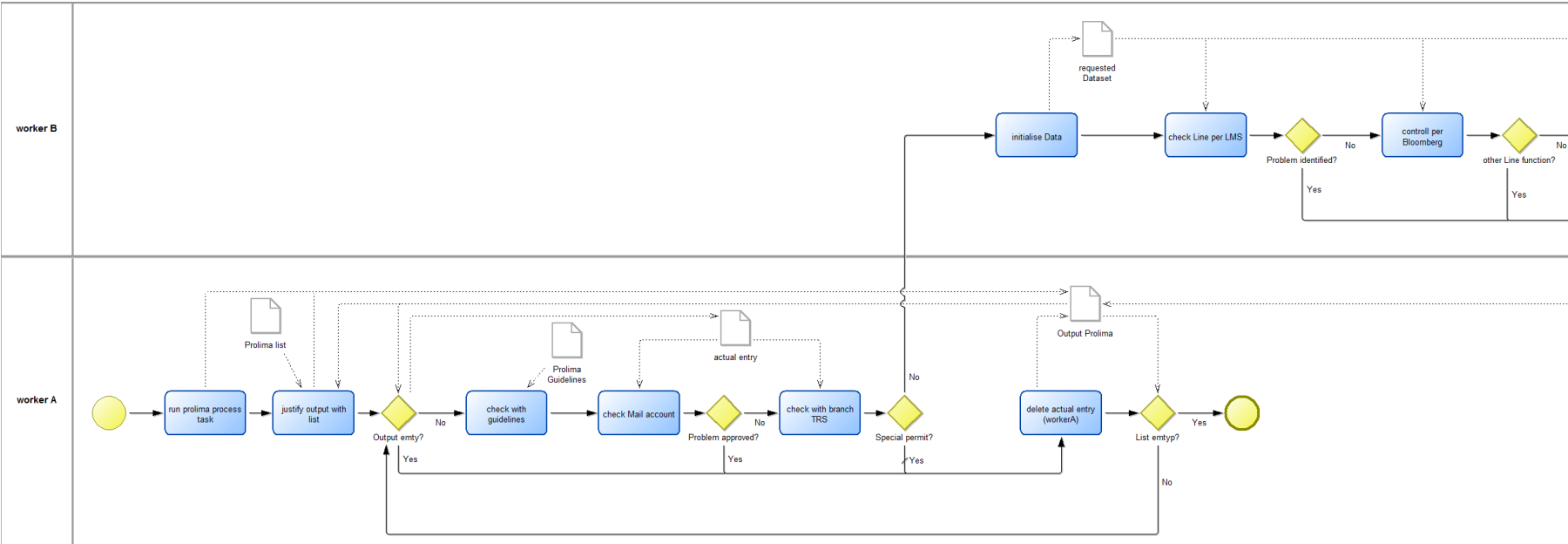


Figure 36: Shows the differences between the original process (picture 30) and the view of employee 2.

Employee 3





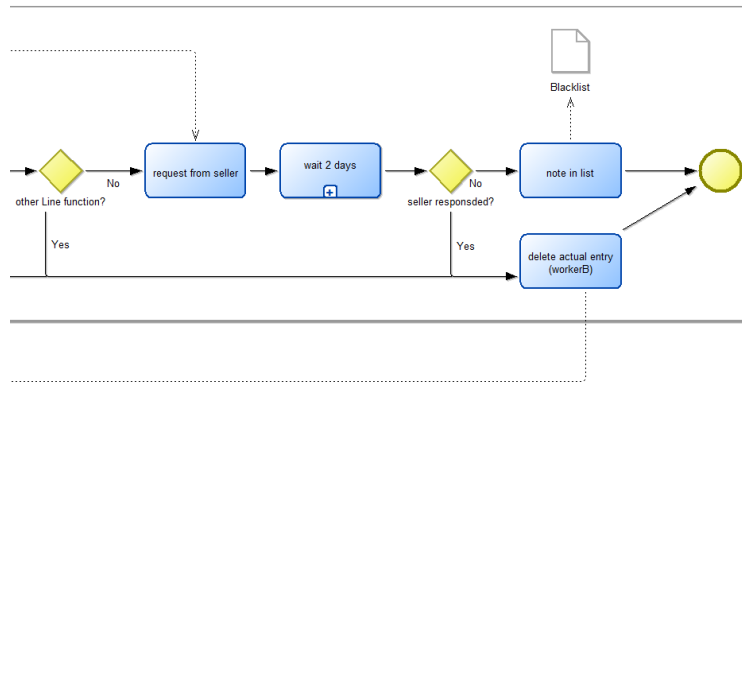
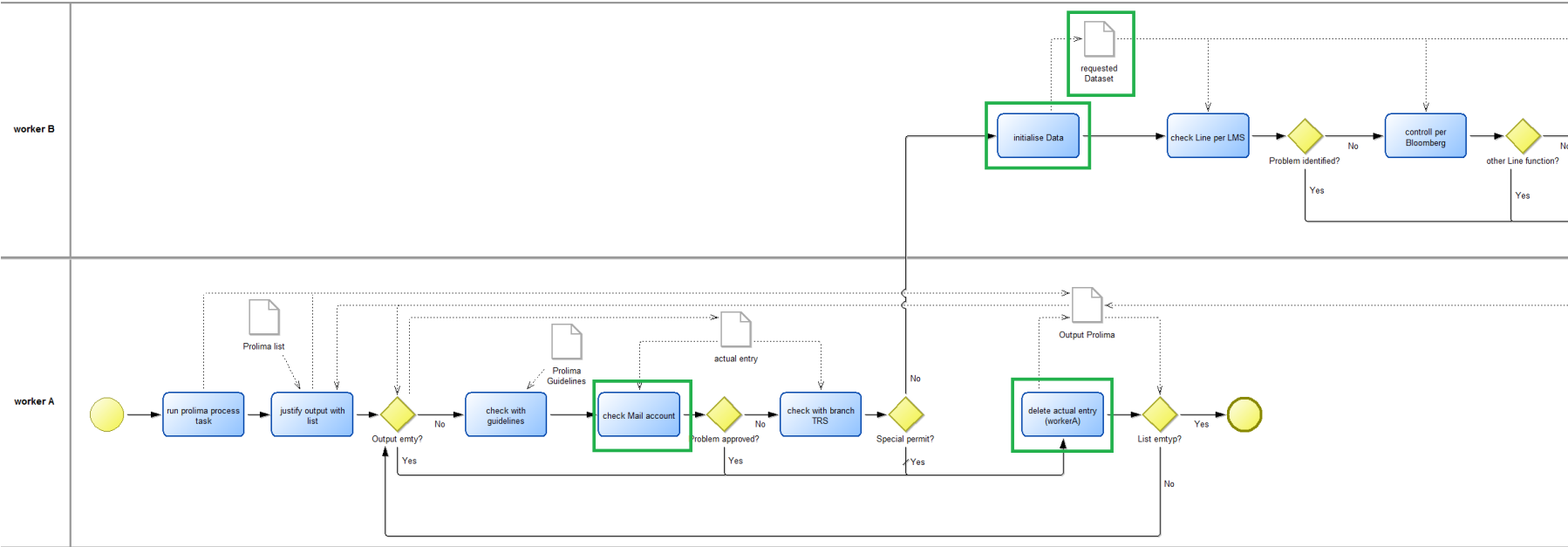


Figure 37: Shows the third Employee of the test process scenario one. Employee 3 view is divided into two parts.

Employee 3 colour



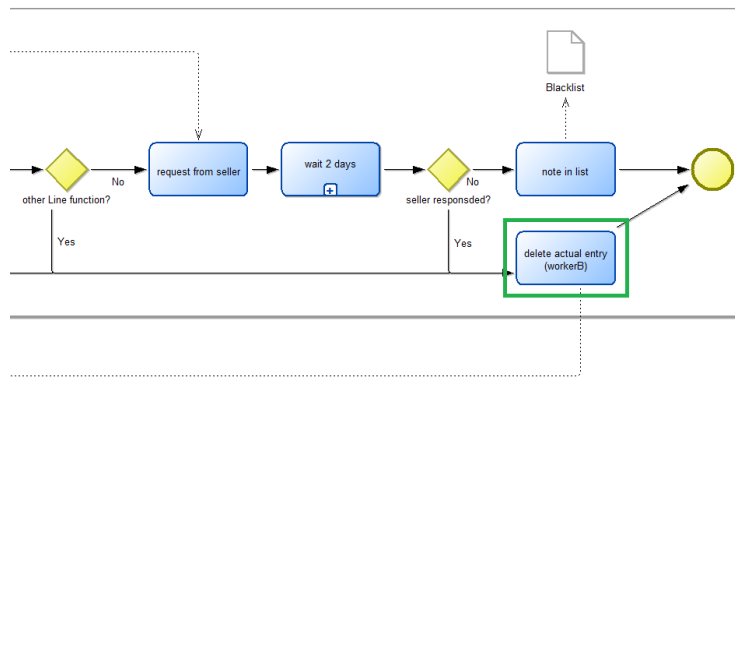


Figure 38: Shows the view of employee 3 with highlighted changes in comparison to the original process (picture 30).

## Trainee

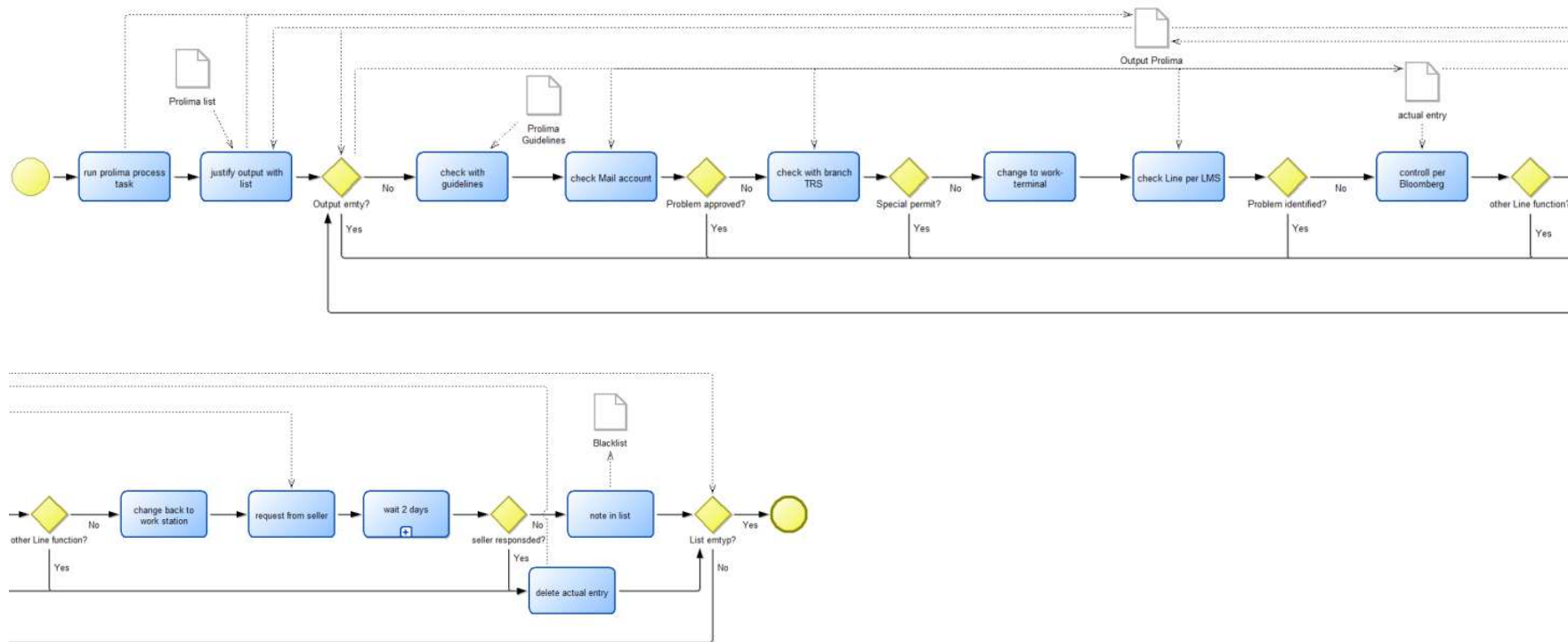


Figure 39: Show the view of a trainee from the department scenario of test process one.

## Trainee colour

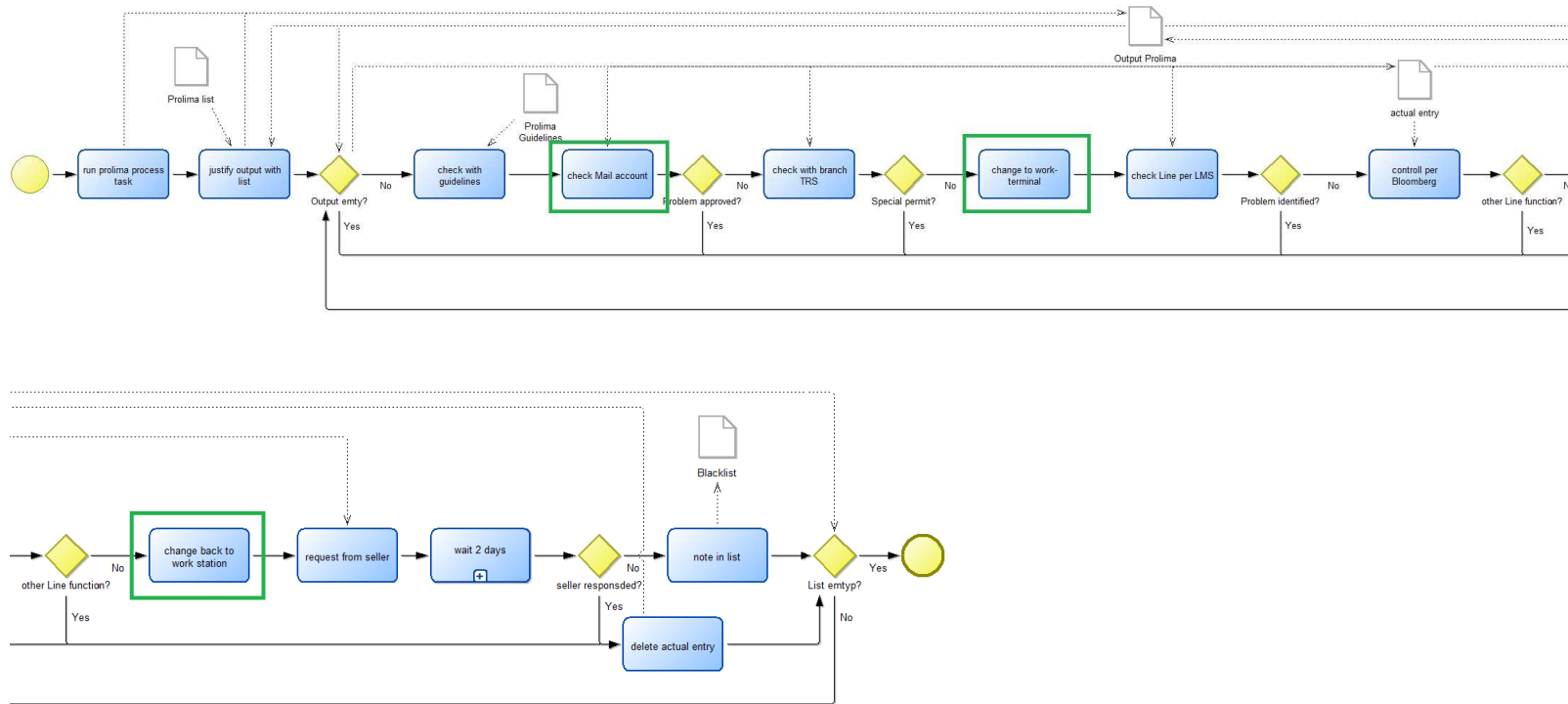


Figure 40: Shows the changes between the original process (picture 30) and the trainee view. The changes are marked with different colours.

## Marginally employed person

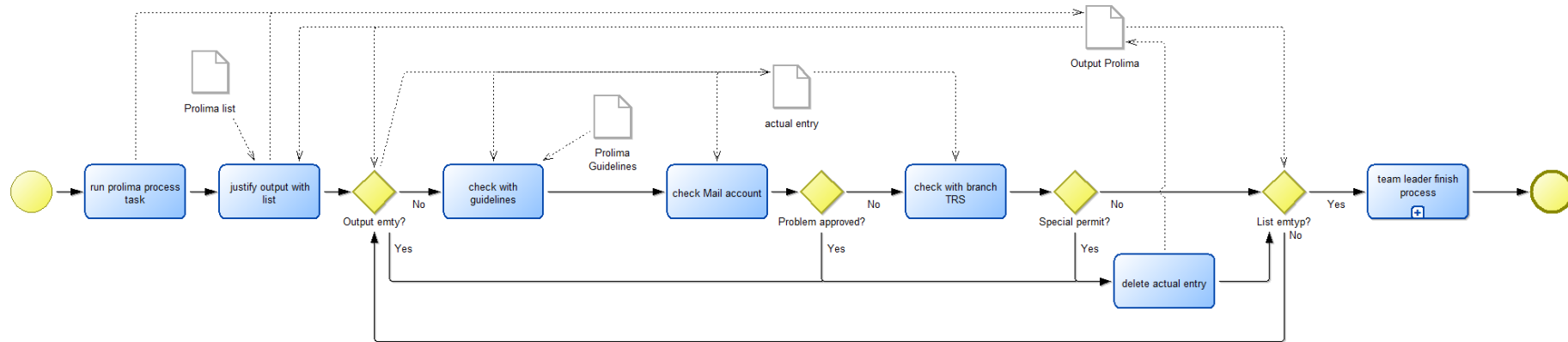


Figure 41: Shows the last view of the test scenario one the Prolima control process. The last person is a marginally employed person.

## Marginally employed person colour

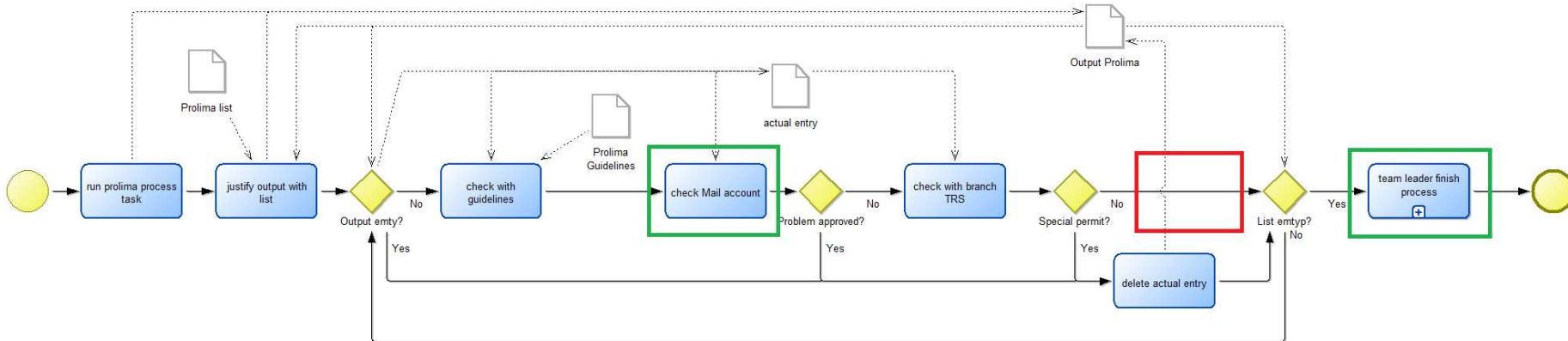


Figure 42: Shows the view of the marginally employed person with marked changes to the original process (picture 30). The marking is done per colouring.

## Merged model

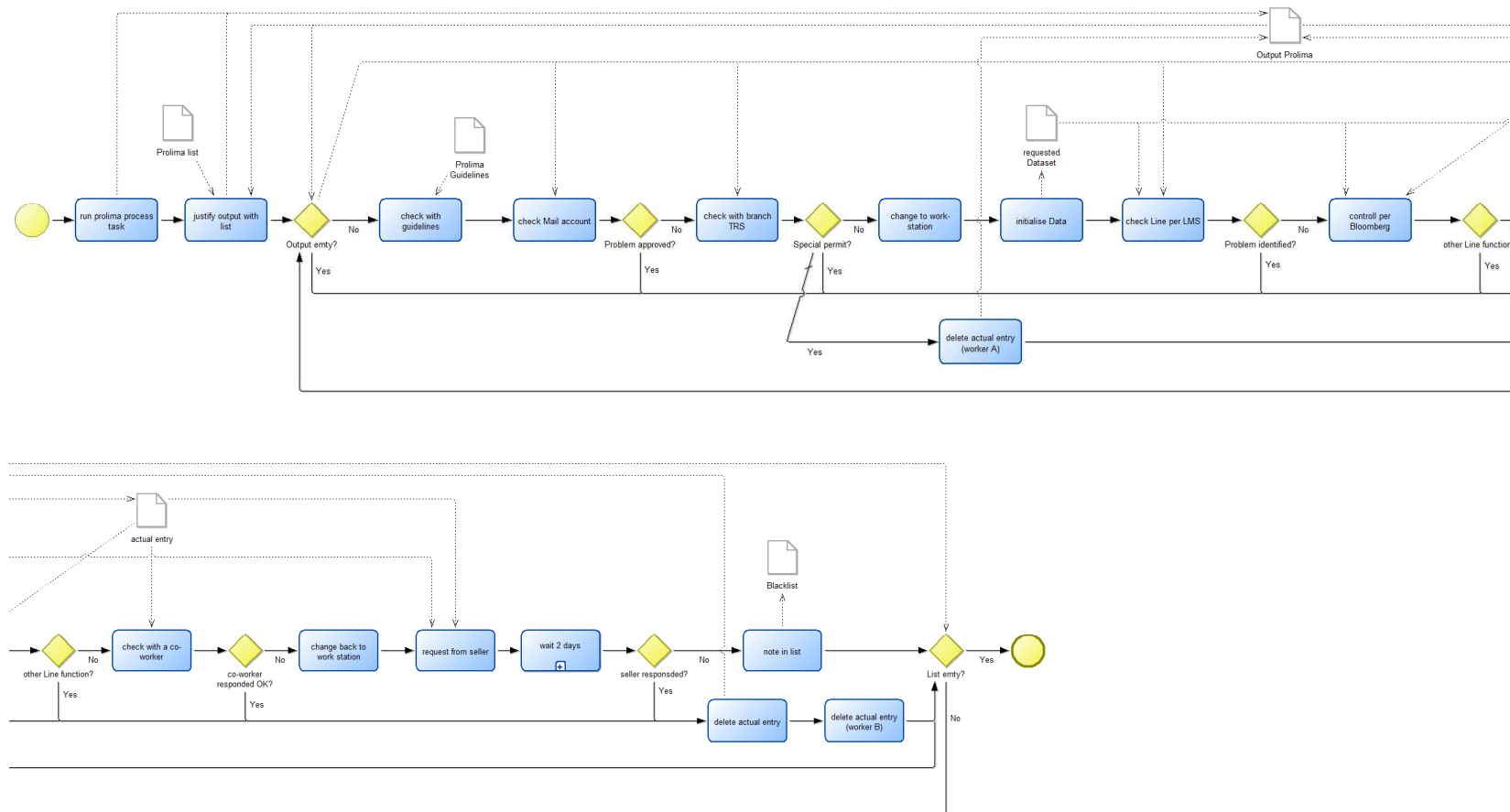


Figure 43: Shows the generated merged model from test process one, the Prolima control process.

## Merged model colour

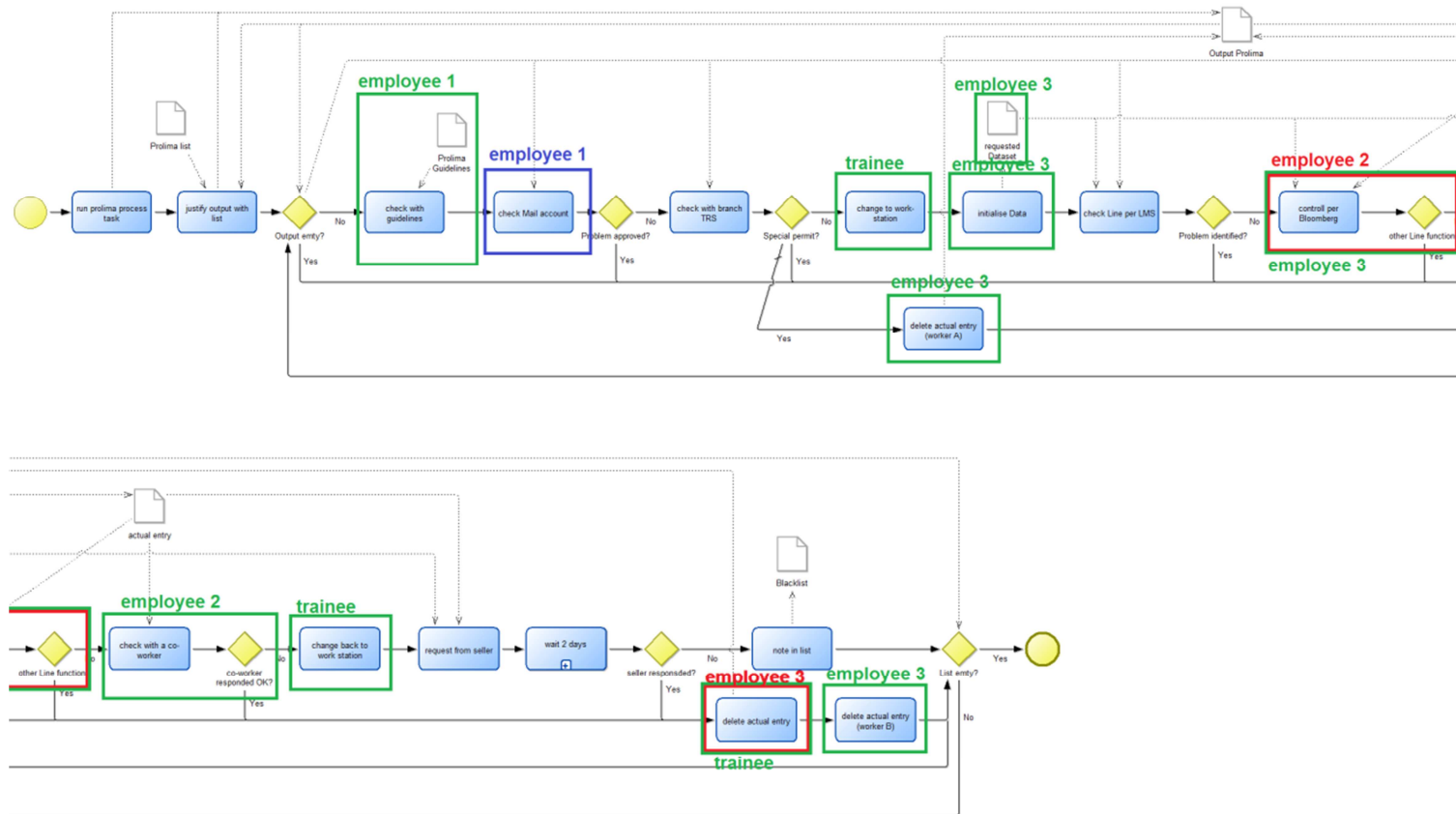


Figure 44: Shows the different changes from the base process to the merged model. Each change is coloured and the source of the change is written over or below the change in the same colour.



## 12.3 University noting process

### Original

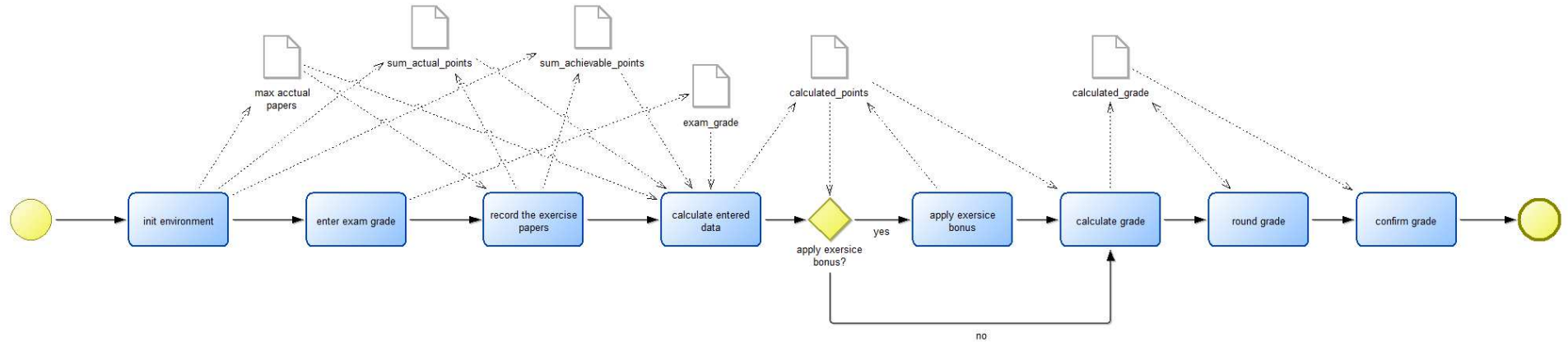


Figure 45: Shows the original process for test process two. The source of this process is a task of an exercise sheet which was adapted to satisfy the criteria for the structure which means no loops.

## Base: Professor

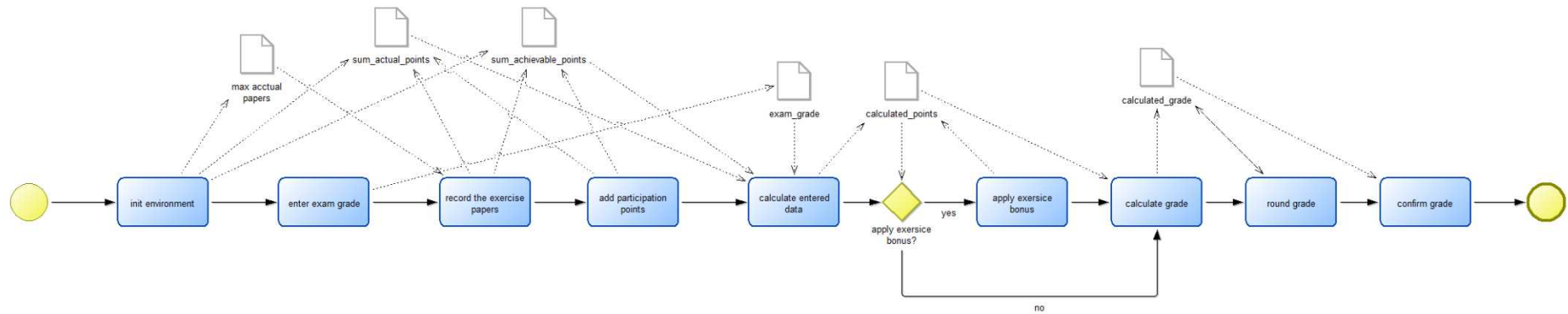


Figure 46: Shows the generated base process view for test scenario two the noting for a university course. This view is owned by the professor.

## Base: Professor colour

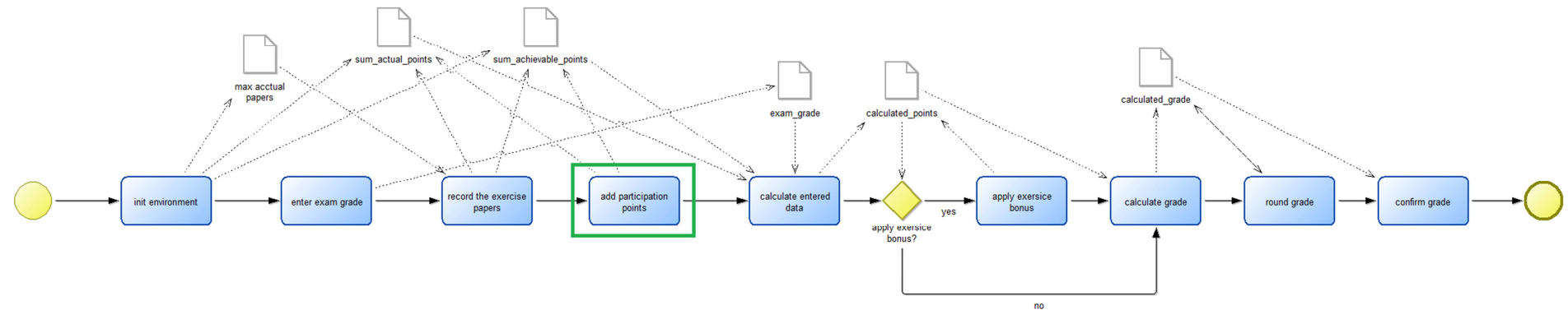


Figure 47: Shows the process view from the professor with per colour marked changes. The setting of the colouring is green for insertion, red for deletion and blue for position change.

## Official lecturer plus second lecturer

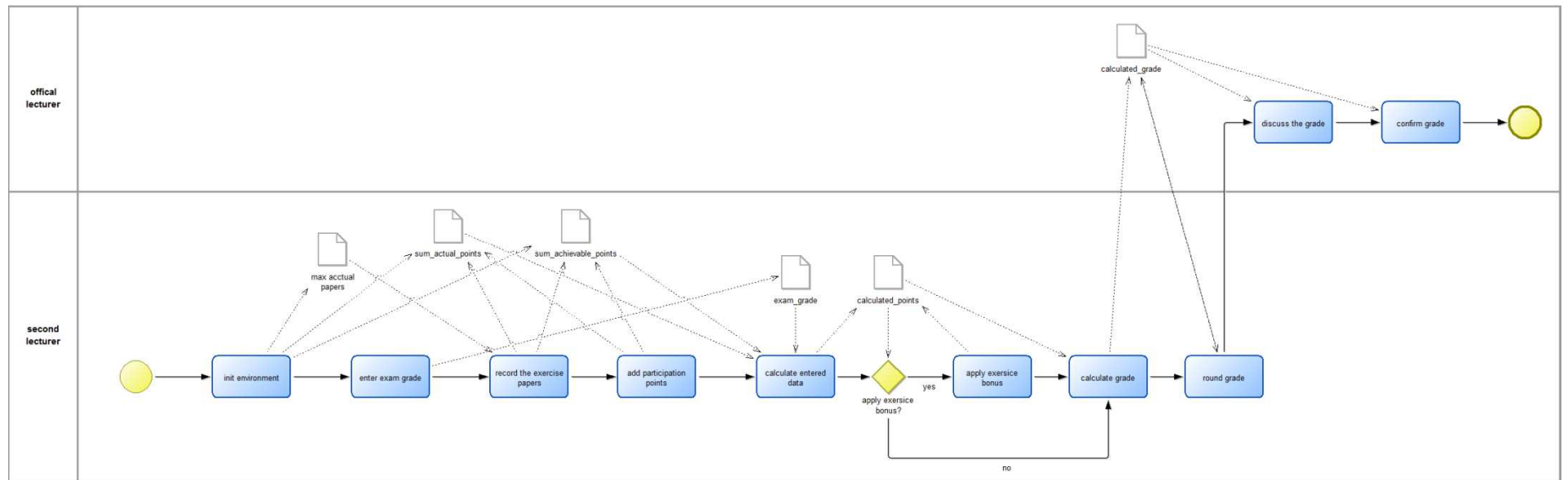


Figure 48: Shows the scenario view of a second lecturer with needing the permission of a supervisor named official lecturer.

## Official lecturer plus second lecturer colour

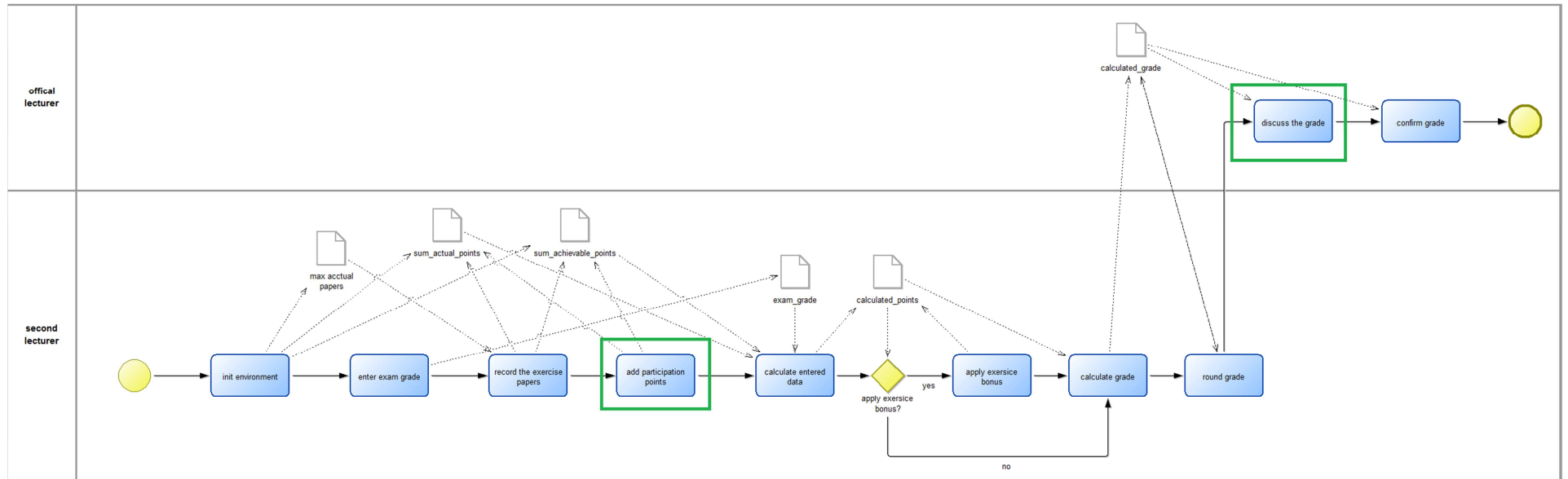


Figure 49: Shows the view of a second lecturer with an official lecturer with coloured changes. The changes are defined as the changes that lead from the original process to this view.

## Second lecturer

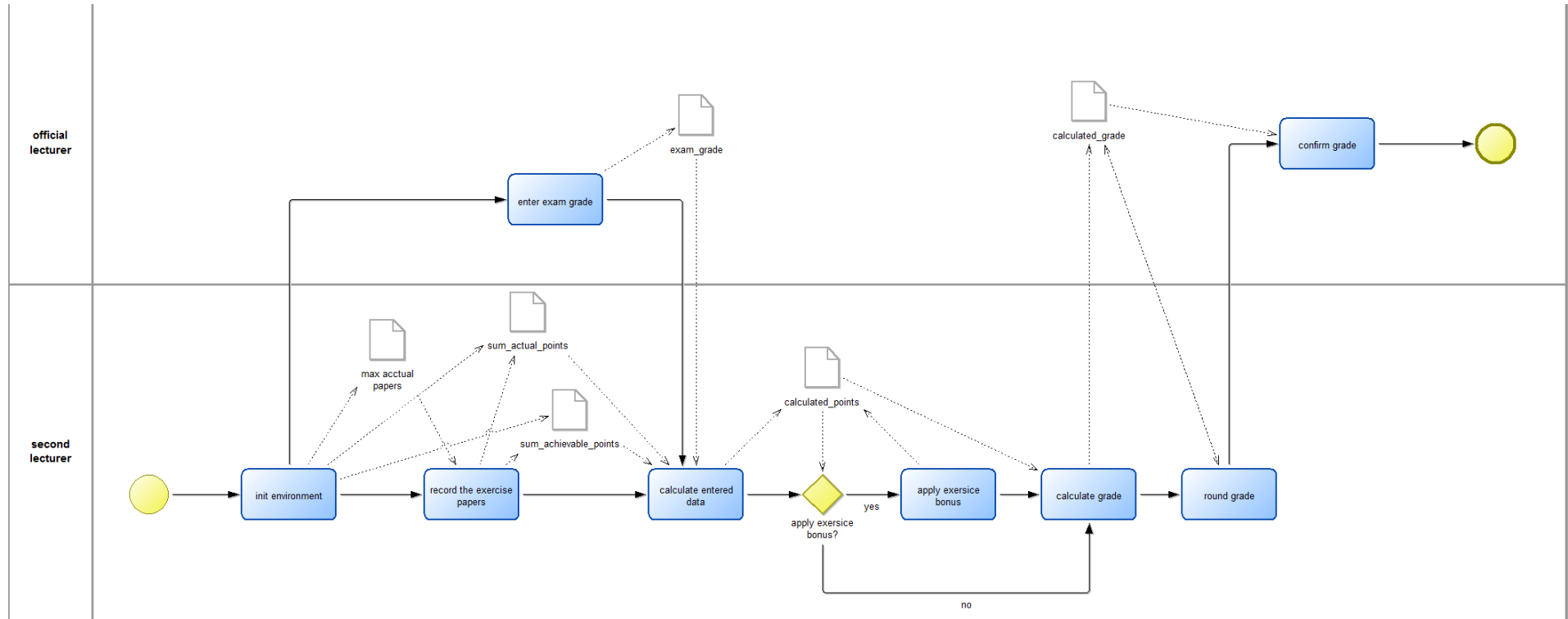


Figure 50: Shows the view of the second lecturer with lower rights than in the view of official lecturer plus second lecturer. In this view an official lecturer is also needed but not only to confirm the note but also to hold the exam and grade it.

## Second lecturer colour

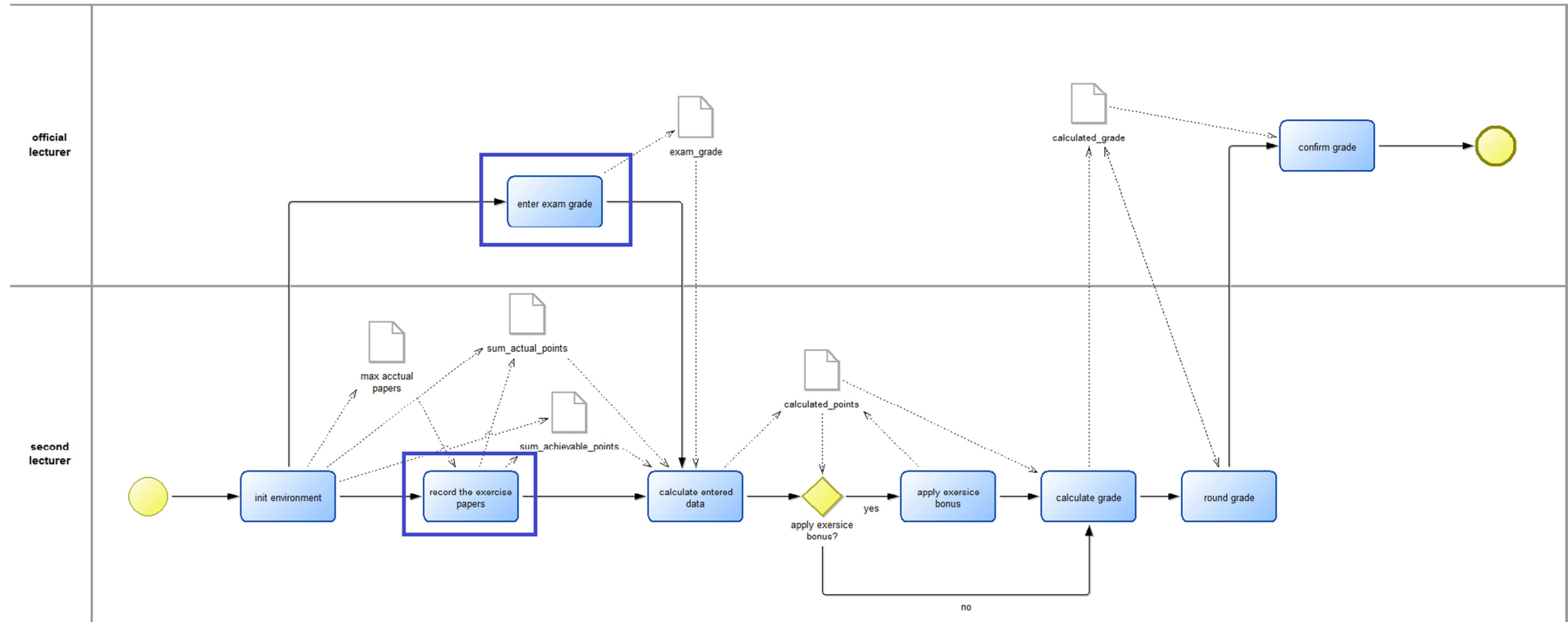


Figure 51: Shows the process of the view from the second lecturer with the official lecturer for grading exam and confirm the end note. The changes which appear between the original process and the view are marked with colour.

## Tutor

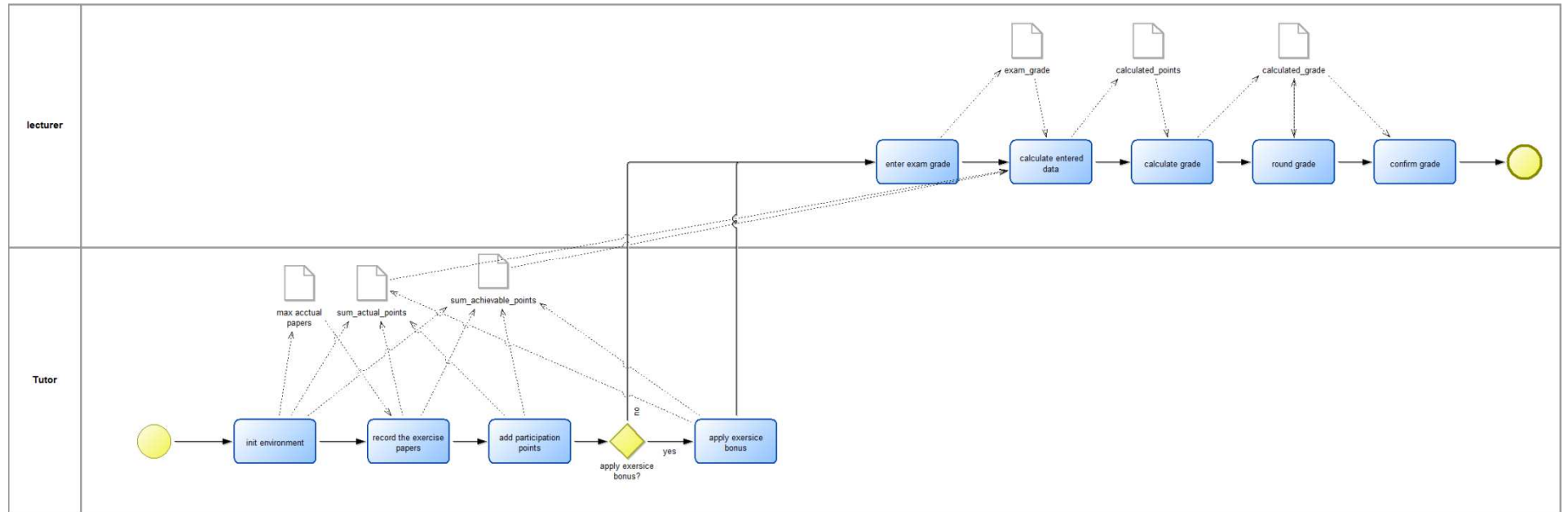


Figure 52: Shows the process view of a tutor. The noting isn't performed by a tutor therefore a second person must be responsible for the noting.

## Tutor colour

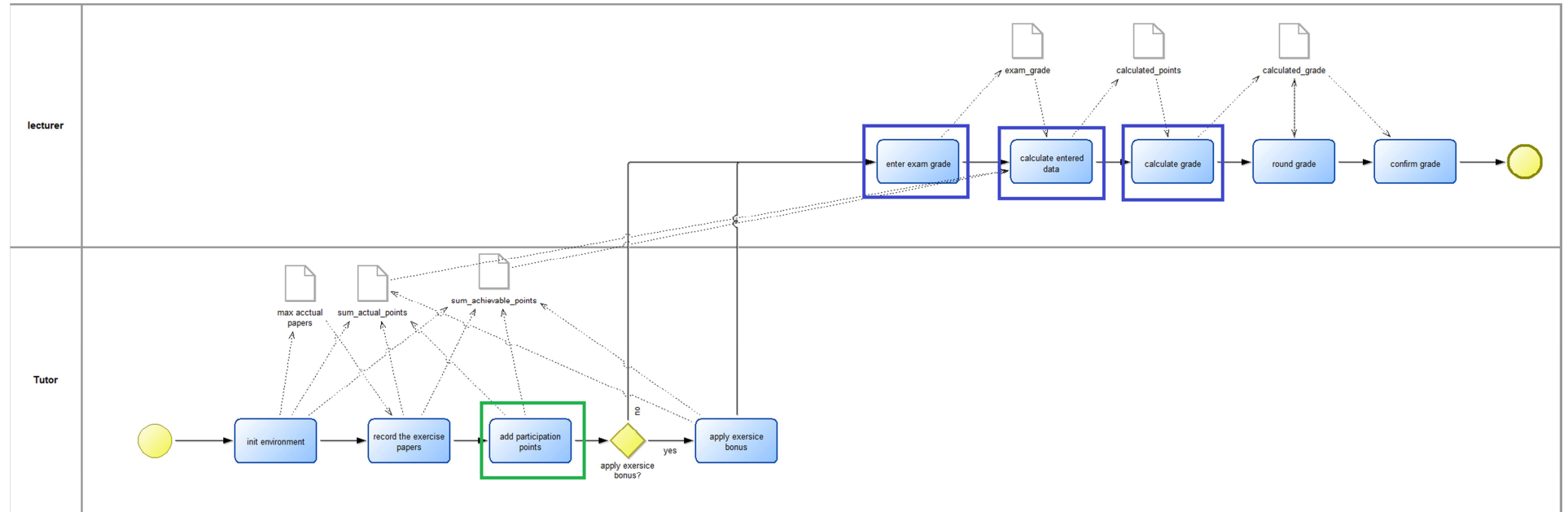


Figure 53: Shows the view of a tutor with marked changes from the original process to this view. The marking is performed per colour.



## Merged model

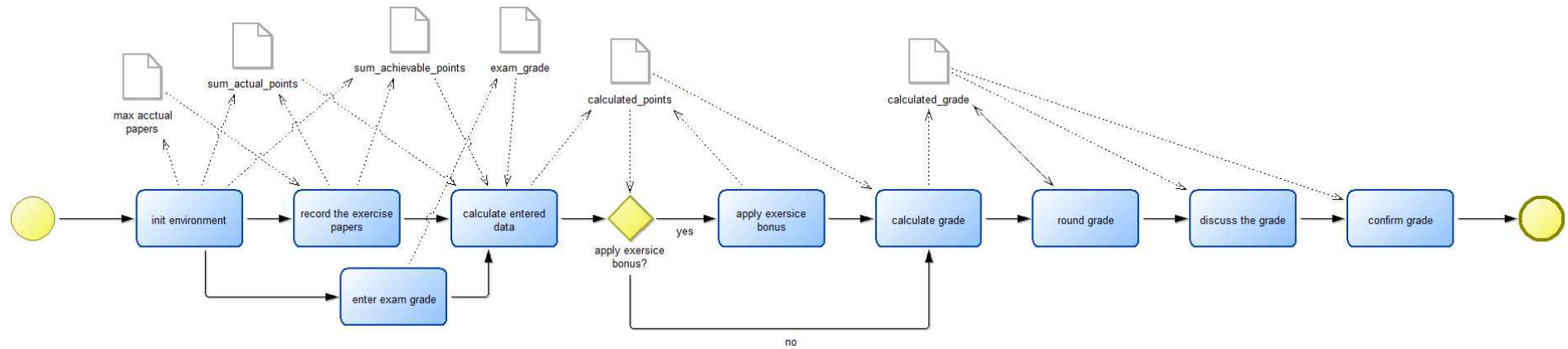


Figure 54: Shows the generated merged model of the noting process for a university note.

## Merged model colour

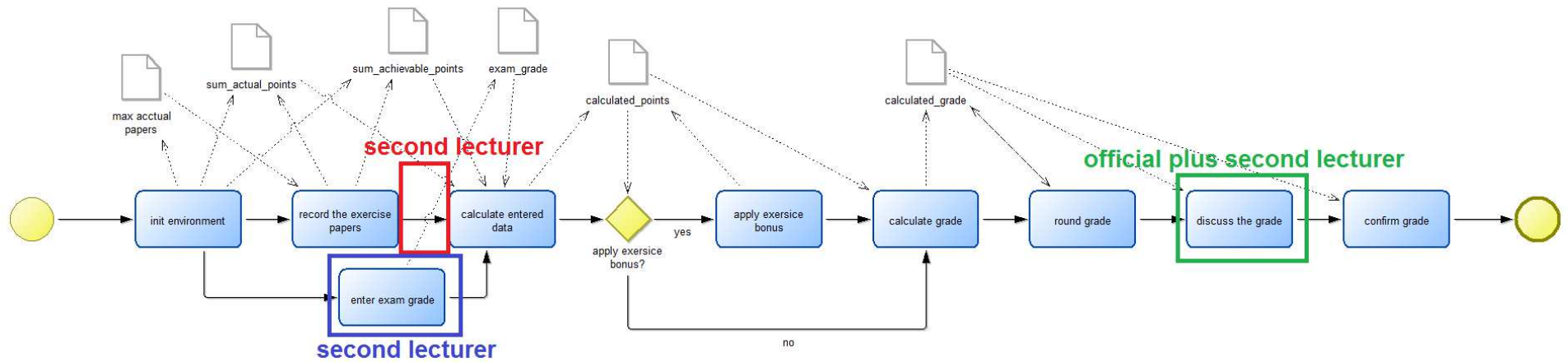


Figure 55: Shows the merged model of test process two the noting of a university course. To show the different changes between the base and the view the changes are marked per colour additional with the name of the view which is the source of the change.

## 12.4 KFZ insurance process

### Original

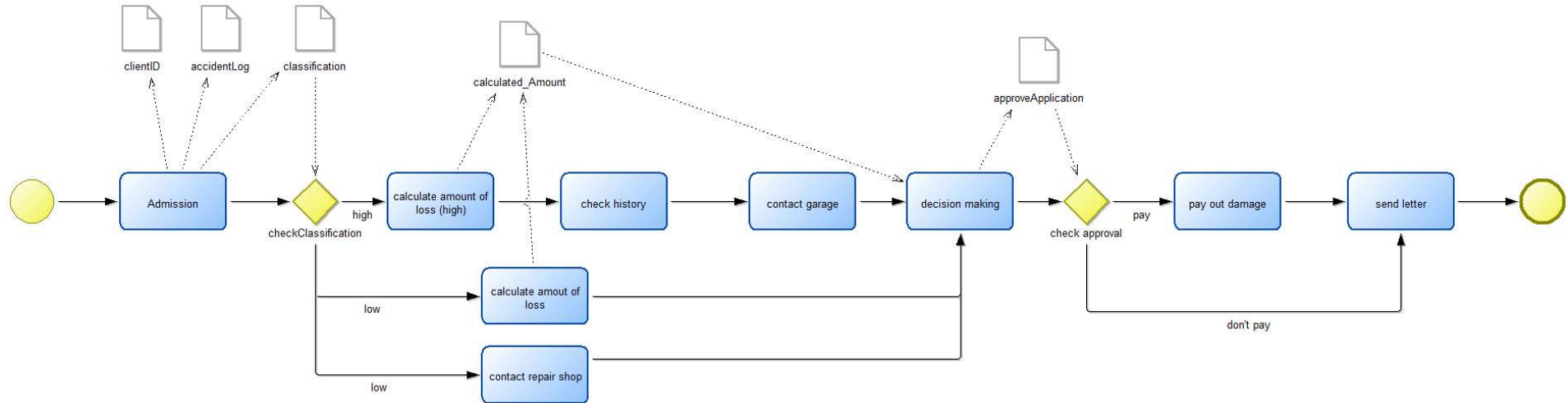


Figure 56: Shows the original process for the last test case number three. The content of this process is a KFZ insurance case. This process is based on a task of an exercise sheet from the university course Workflow Technologies of the University of Vienna.

## Base: Adviser

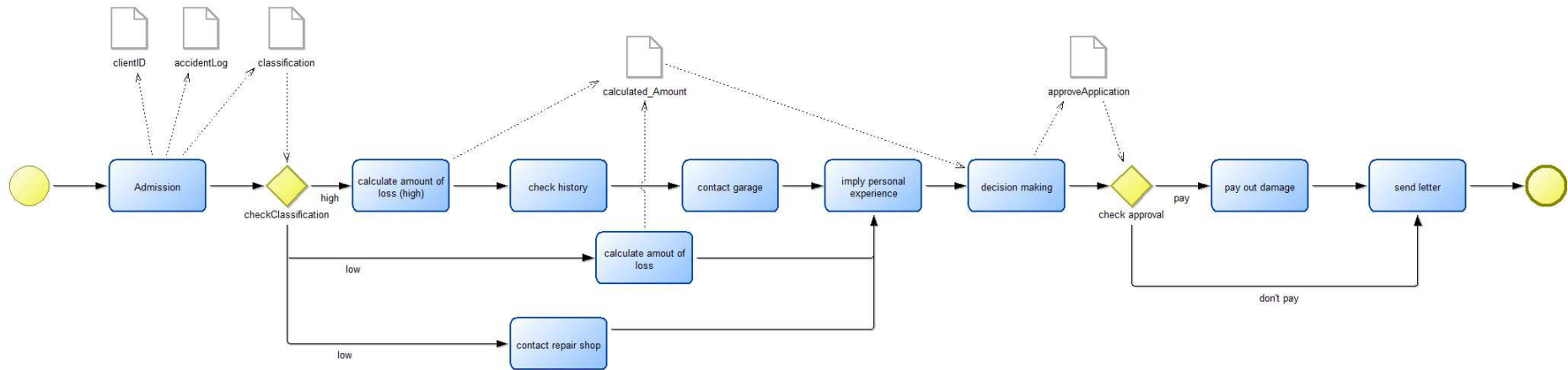


Figure 57: Shows the base process of the generated scenario for the test process KFZ. The owner of this view is an Adviser with full rights.

Base: Adviser colour

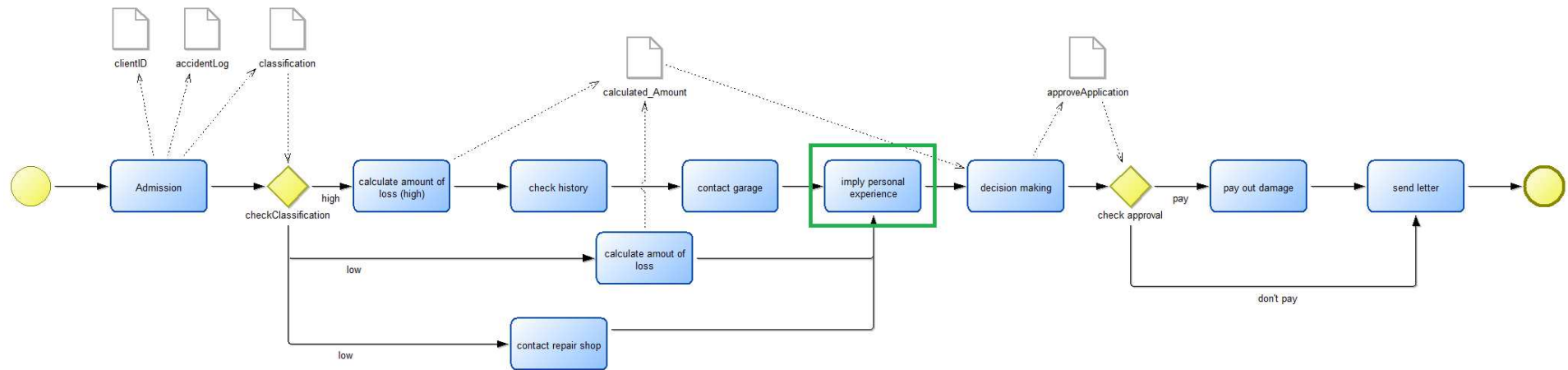
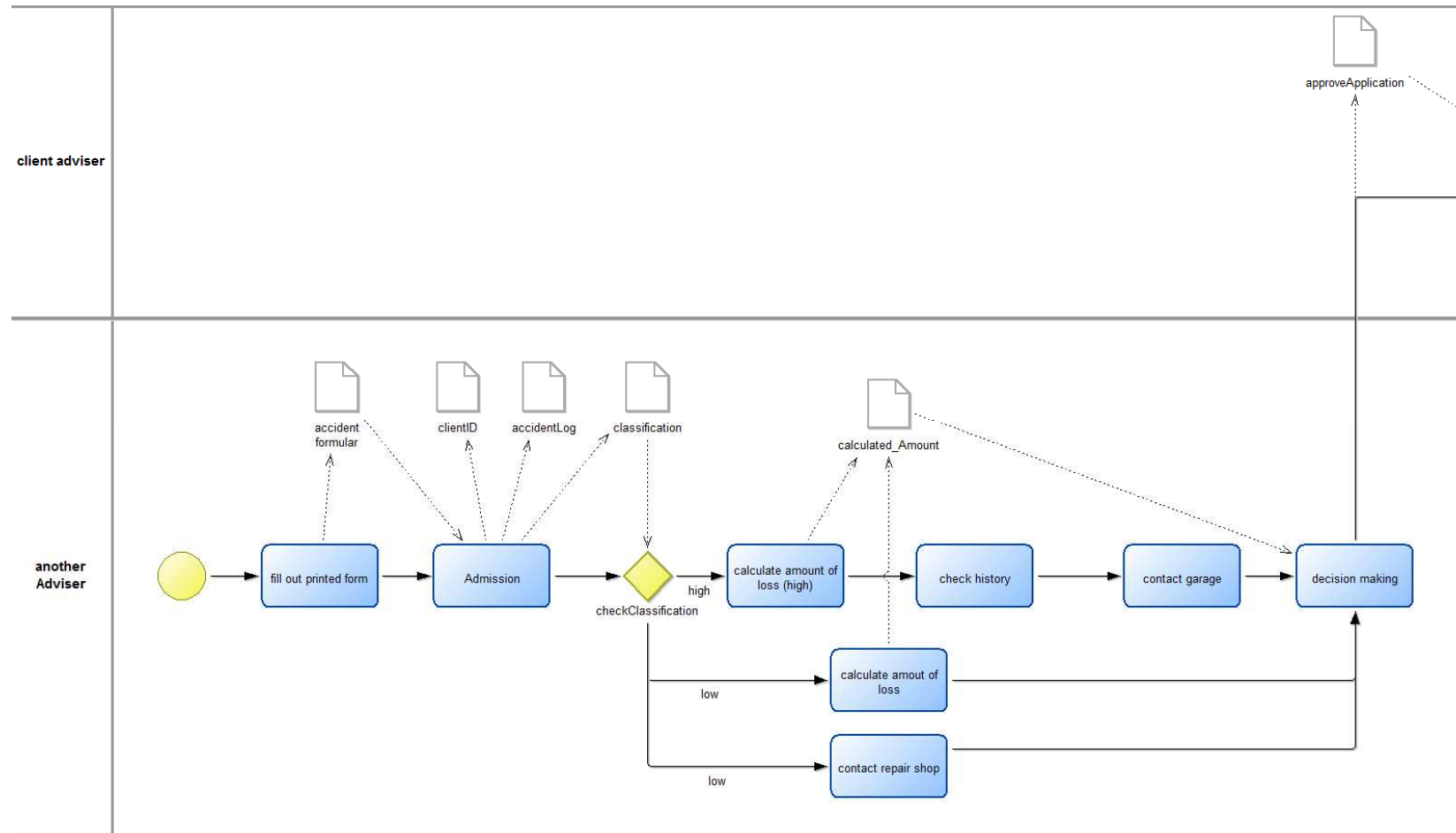
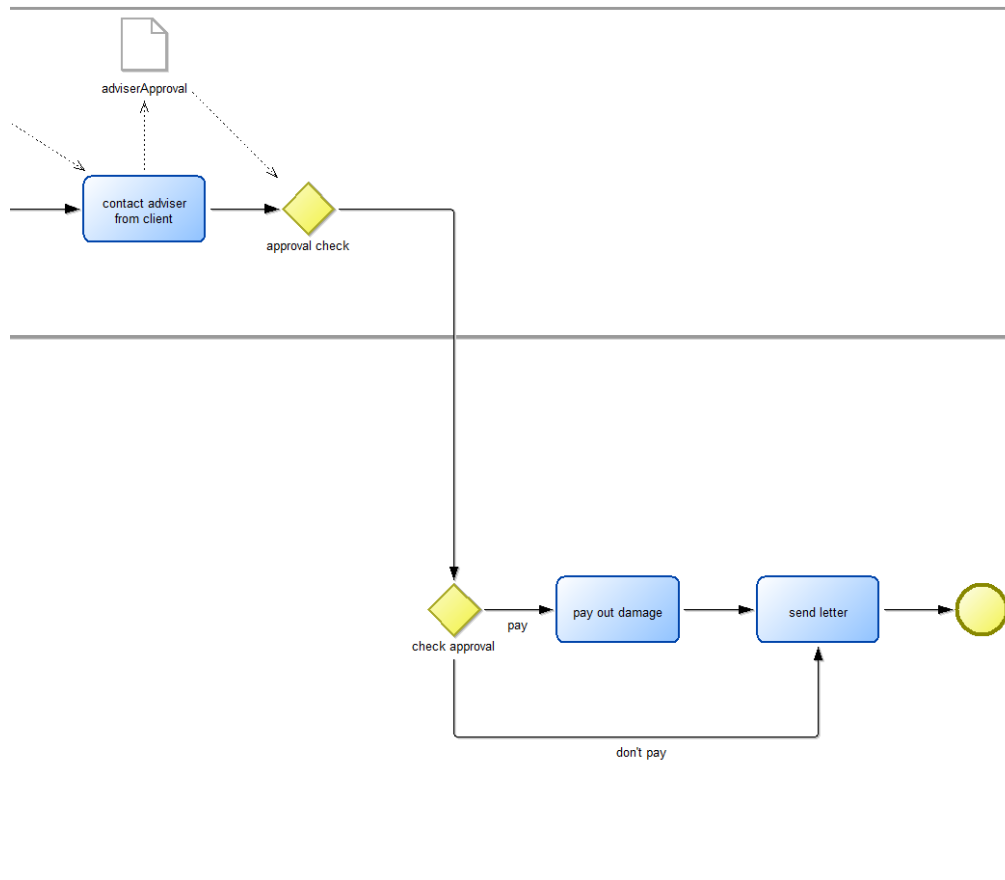


Figure 58: Shows the base process with highlighted changes to the original process. The highlighting is executed per colour. The meanings of the different colours are: green for insertion, red for deletion and blue for position change.

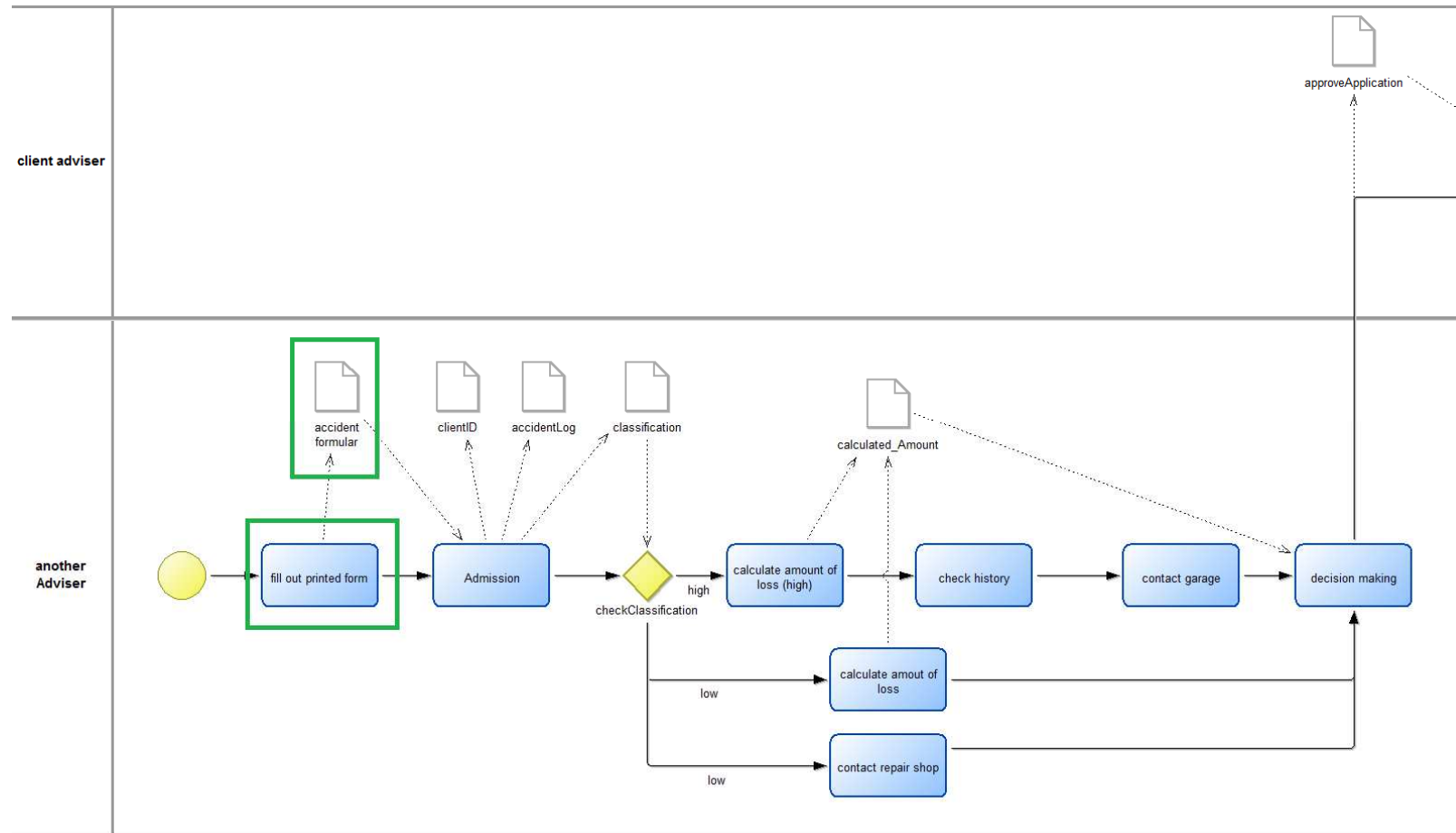
## Adviser





*Figure 59: Shows the view of another adviser than the base process. The view of this scenario is based on a scenario from real life where the real adviser of a customer is on holiday. This process is made from the substitute advisor which is the owner of this view who needs to get the approval of the real adviser.*

## Adviser colour





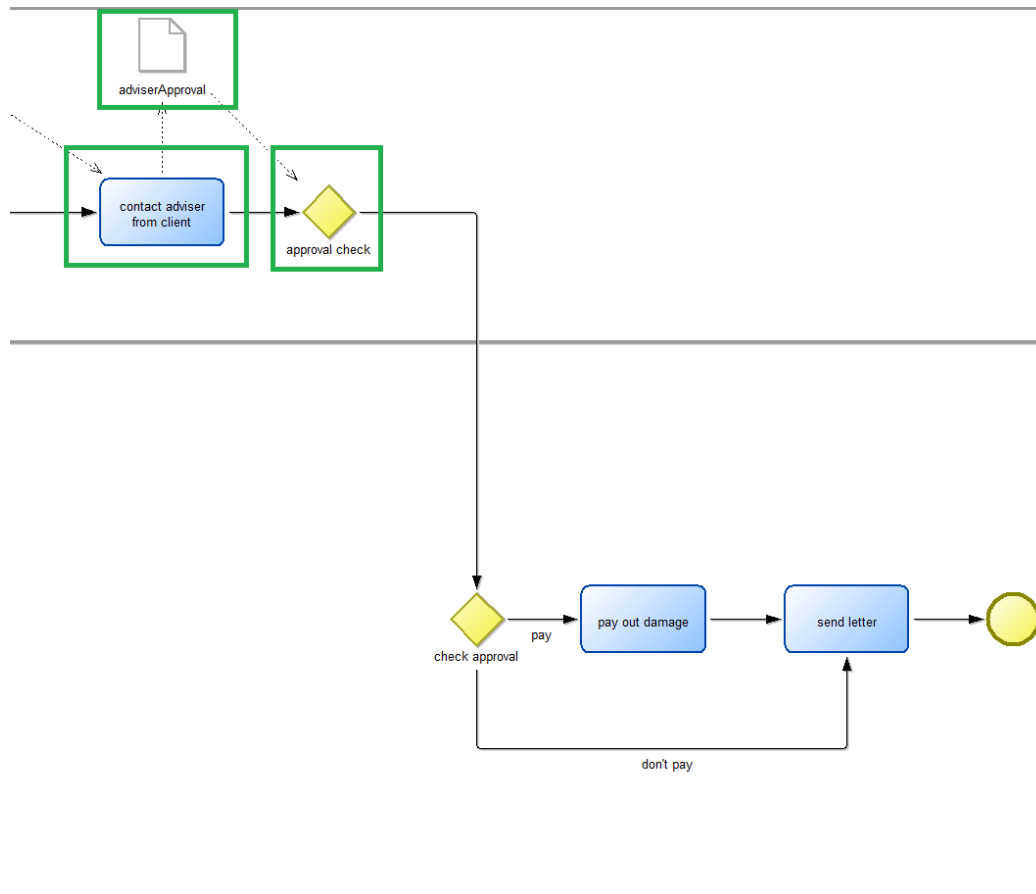
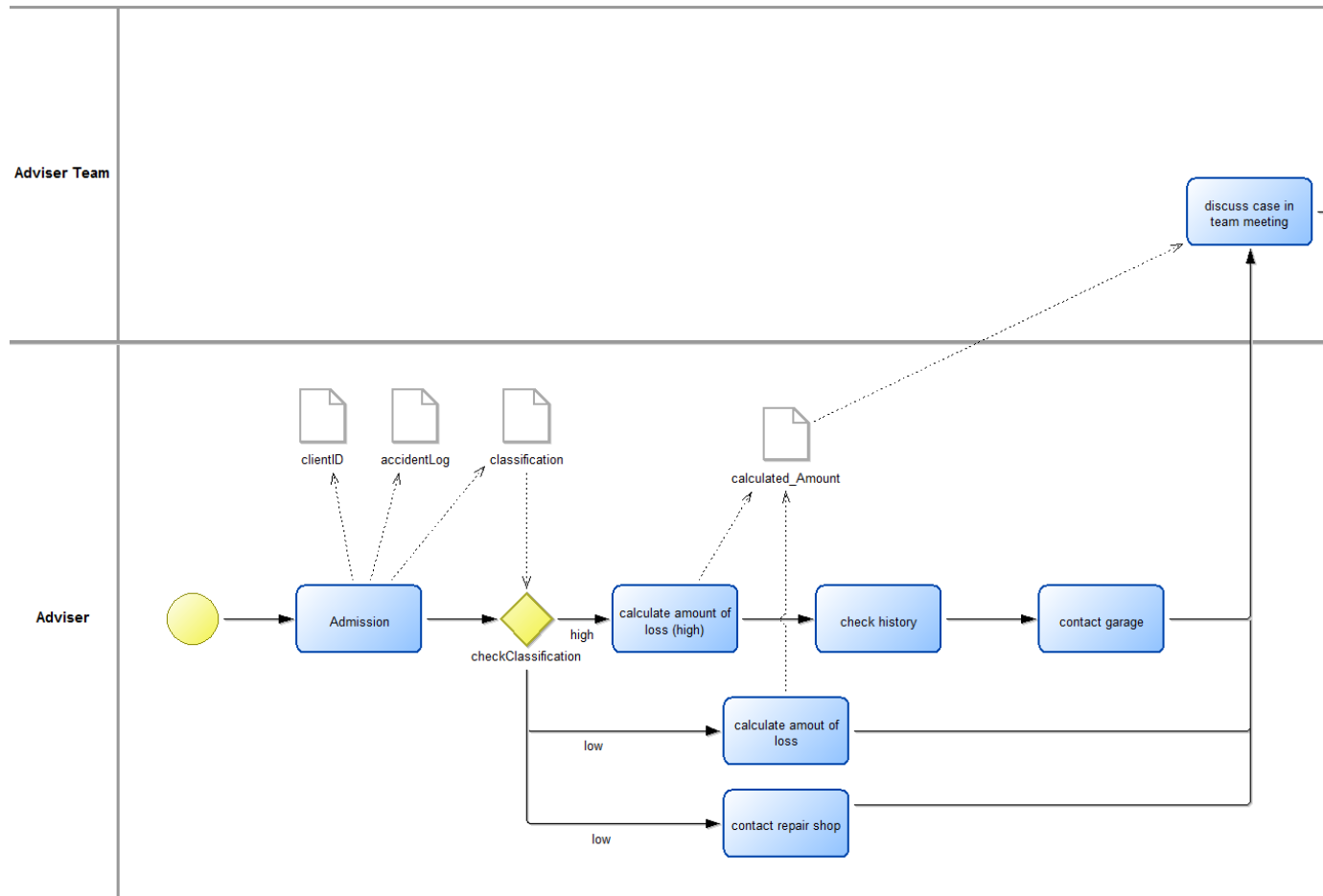


Figure 60: Shows the view of an adviser which is based on the scenario that two advisers are needed for this case because of a holiday. The differences between the original process and this view are marked with colour.

## Team adviser



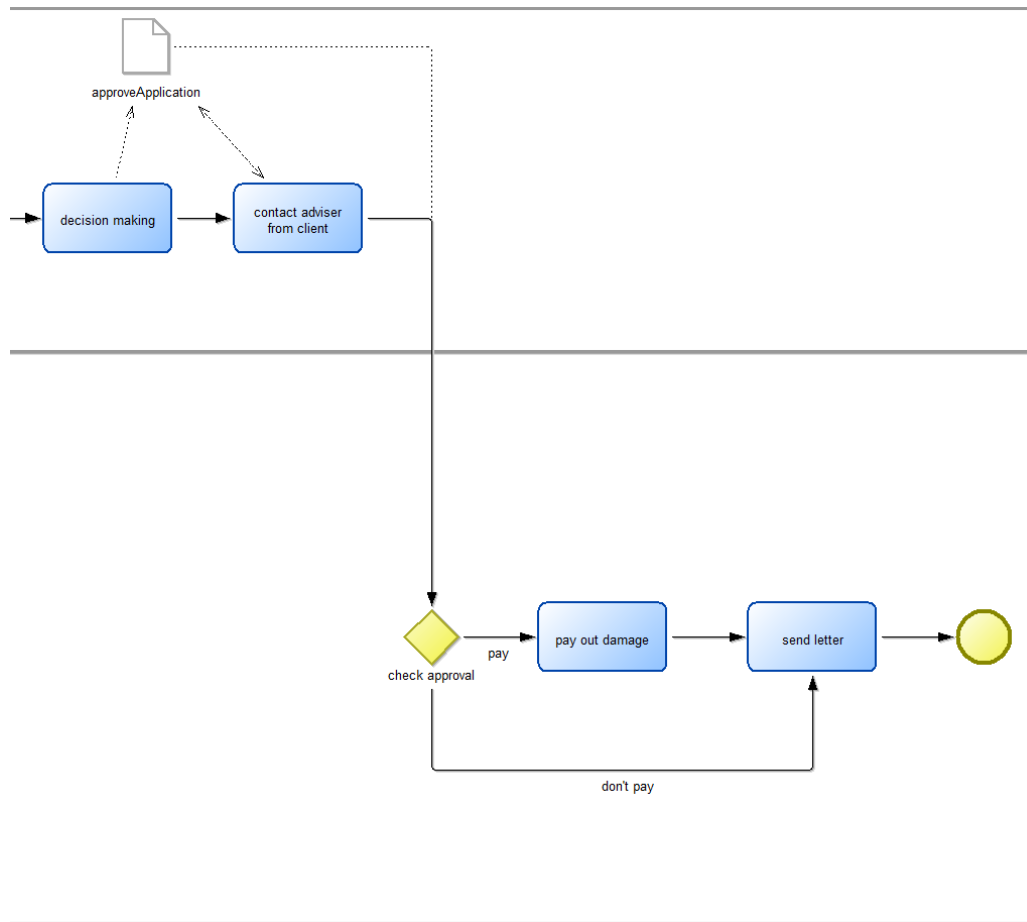
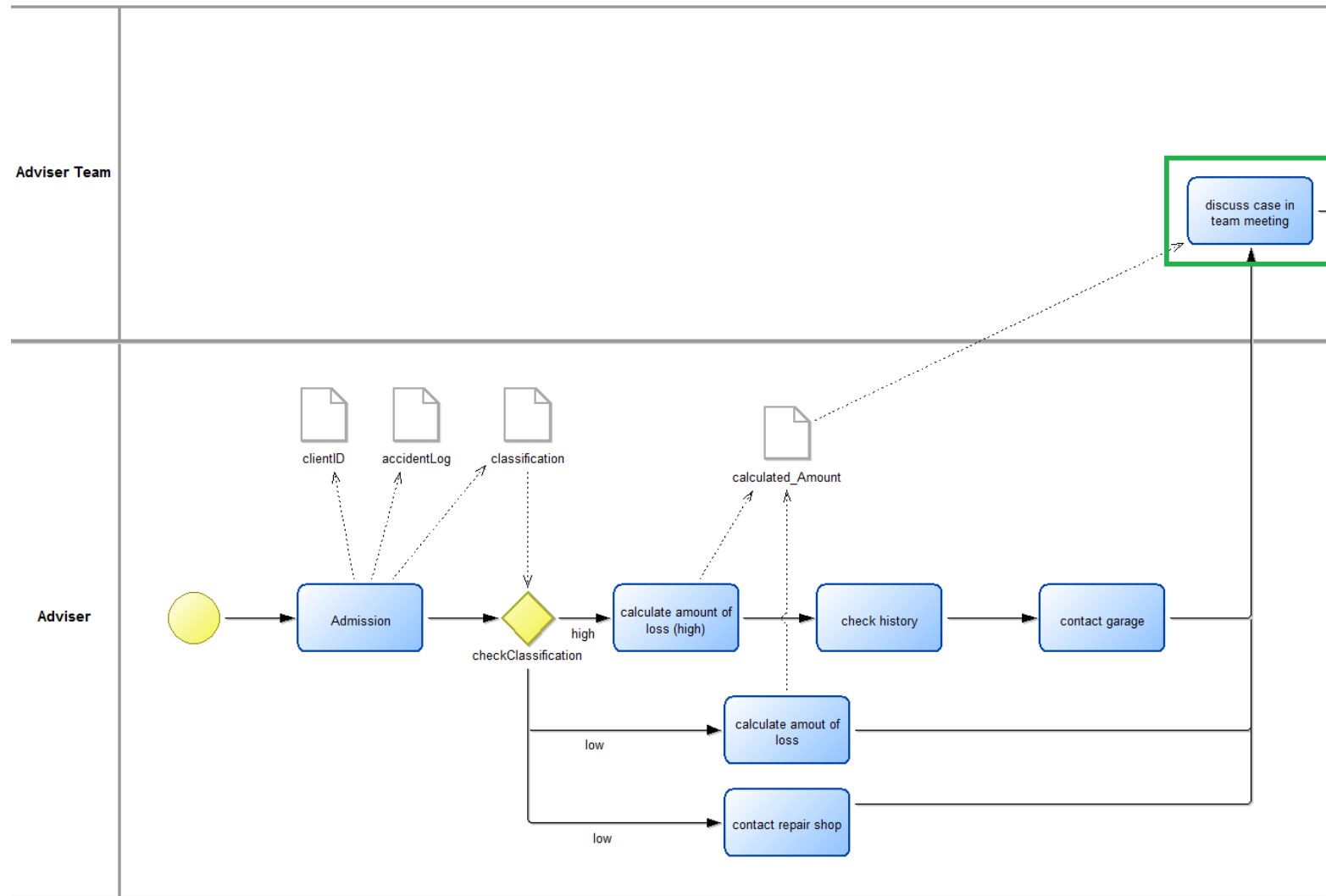


Figure 61: Shows another real life scenario which shows that the case is processed from one adviser but the final decision is made by a whole team.

## Team adviser colour



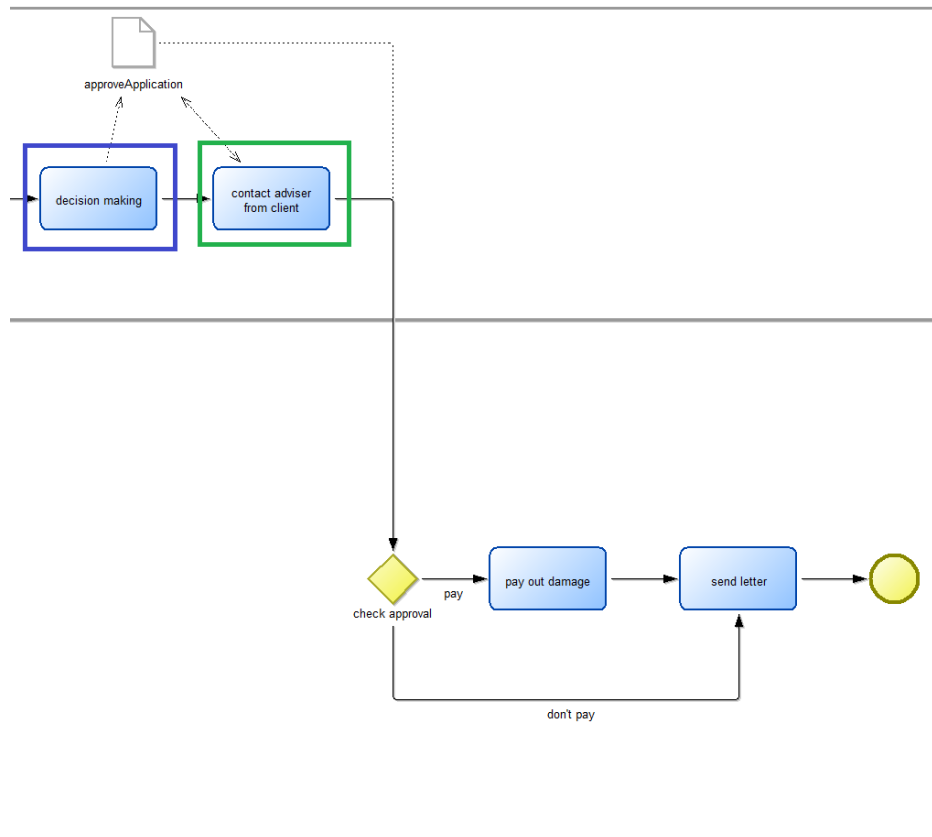


Figure 62: Shows the view of the team adviser with marked differences from the original process to the view. The marking is processed with colour.

## Learning adviser

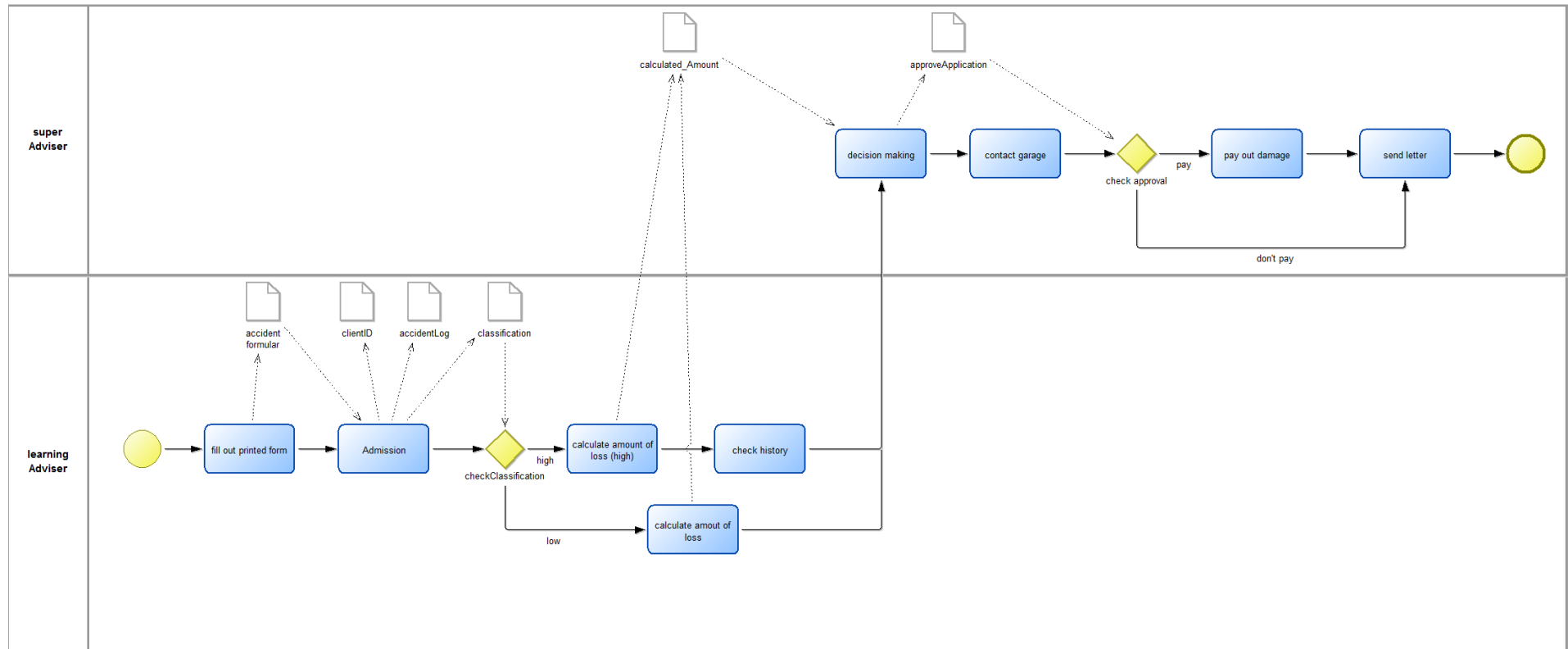


Figure 63: Shows the process of a learning adviser which holds a lot of rights but needs permission and support from a teaching adviser.

## Learning adviser colour

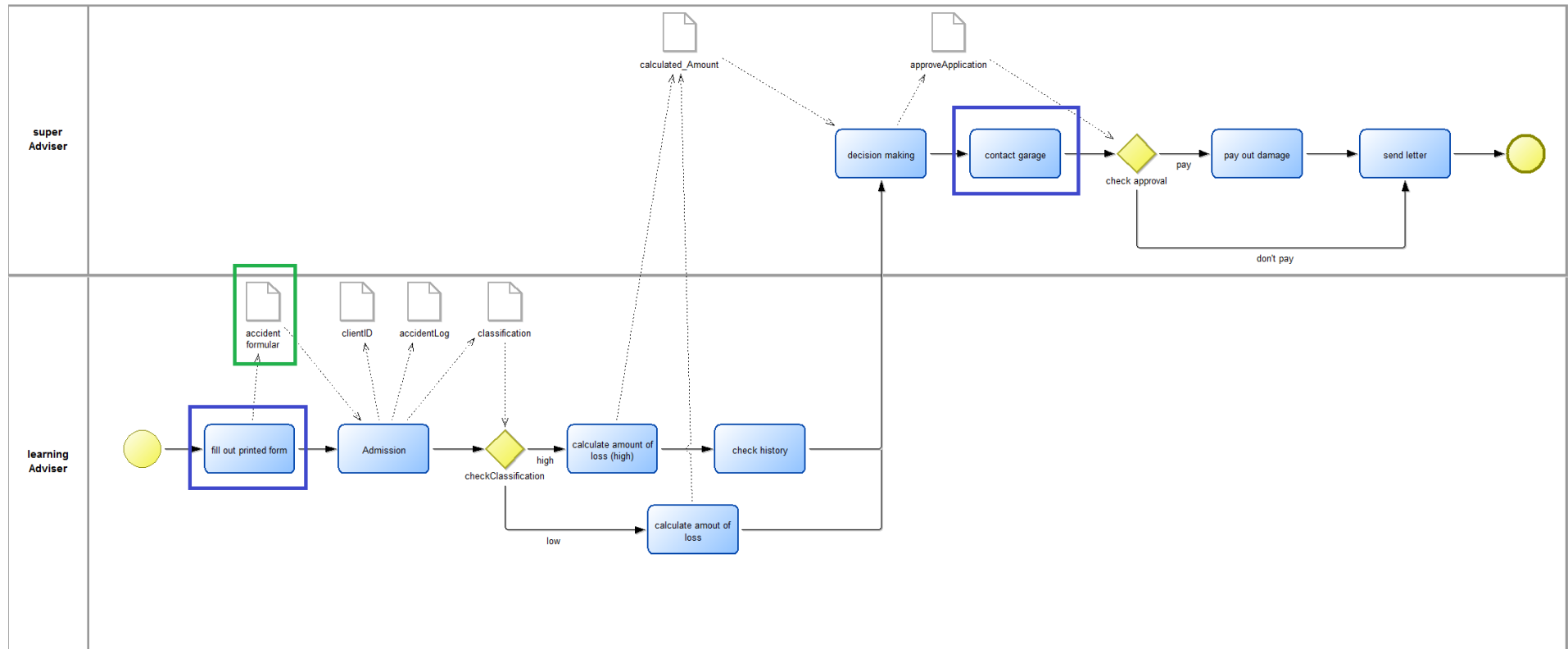
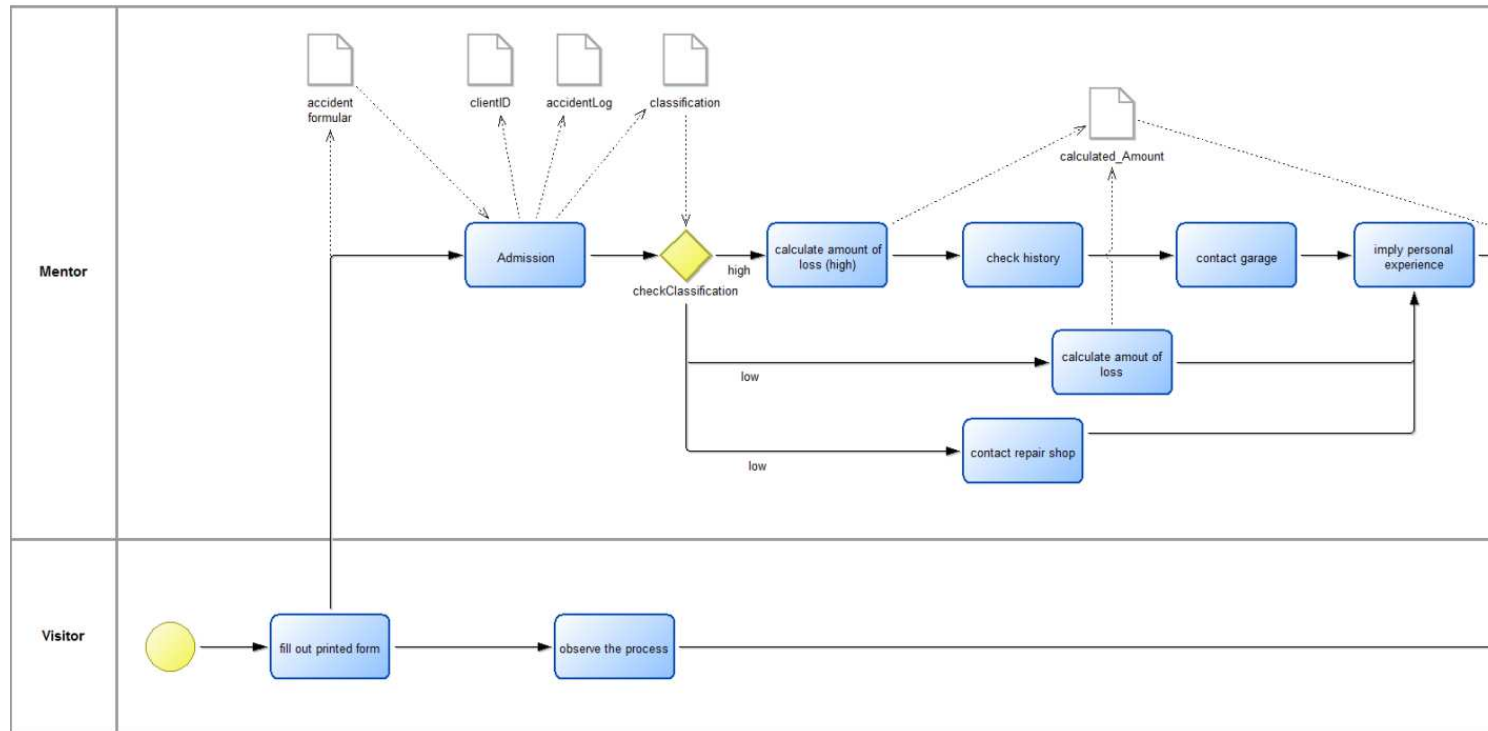


Figure 64: Shows the process of a learning adviser with coloured changes. The changes represent the way how to get from the original process to this view.

## Visitor





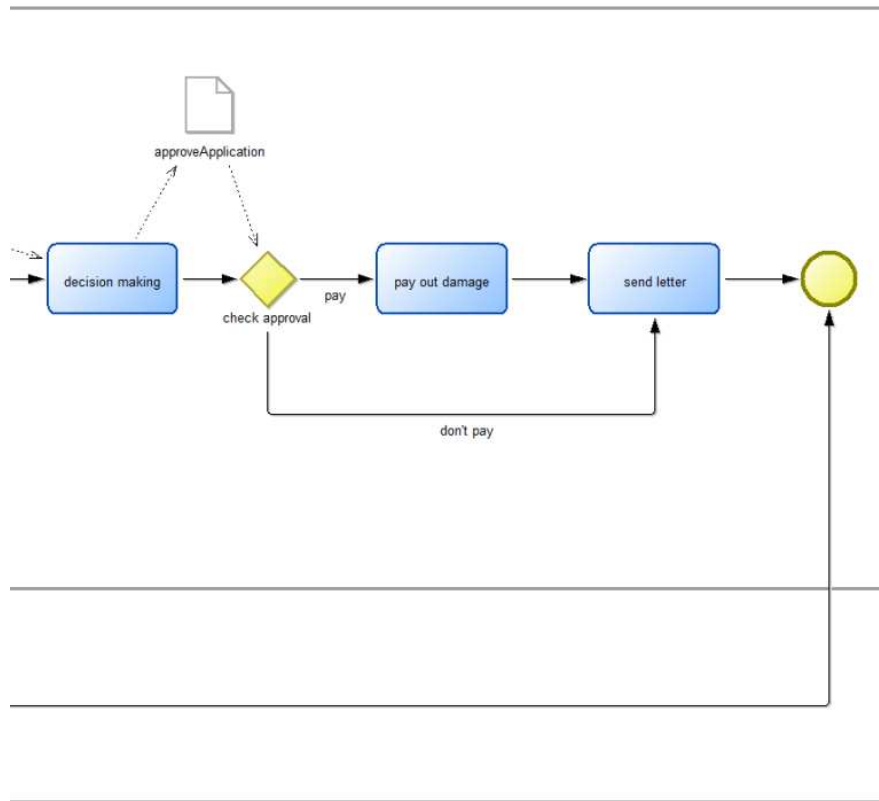
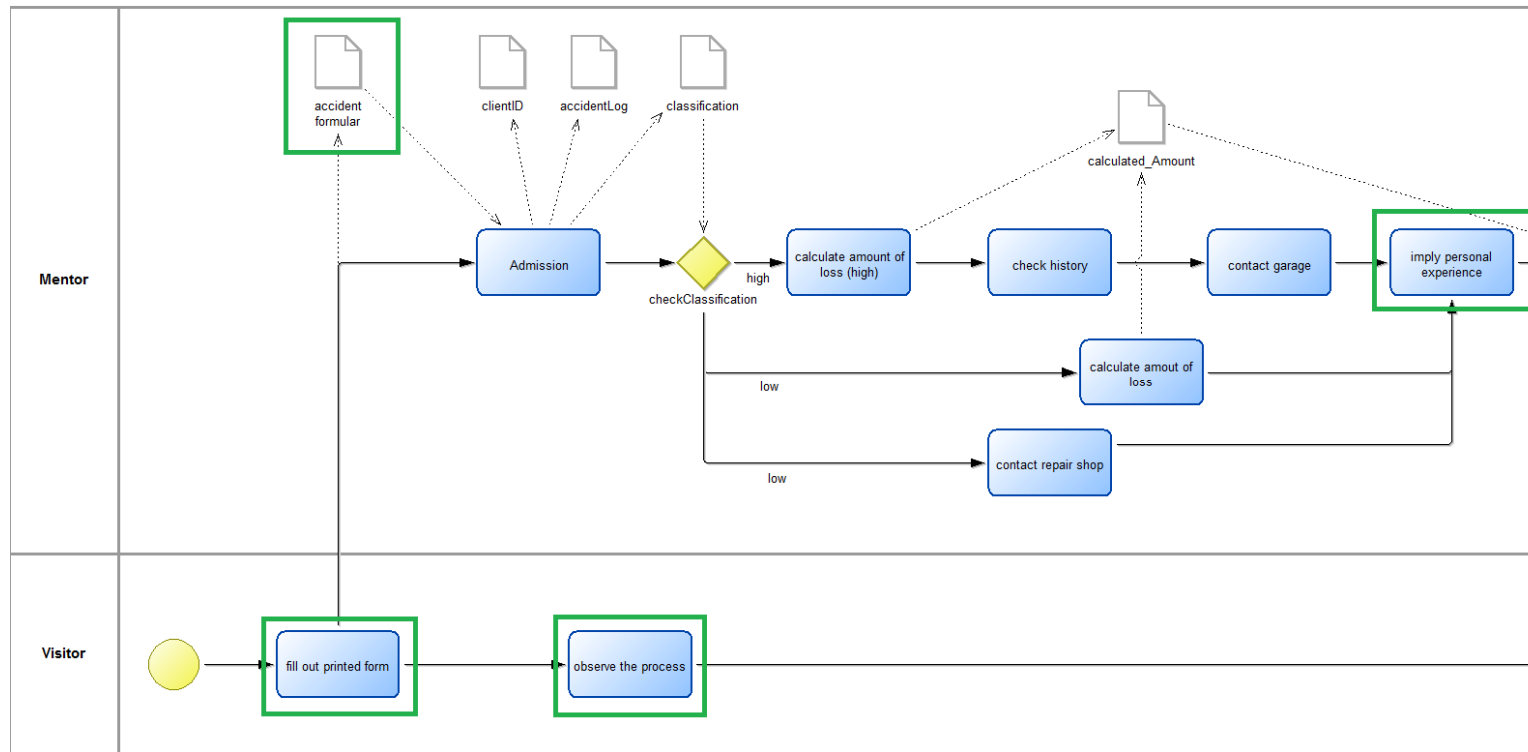


Figure 65: Shows the case of a visitor which can happen when an employee of another department makes a further education in this department. The whole process is executed by a different person but the visitor observed the whole process.

## Visitor colour



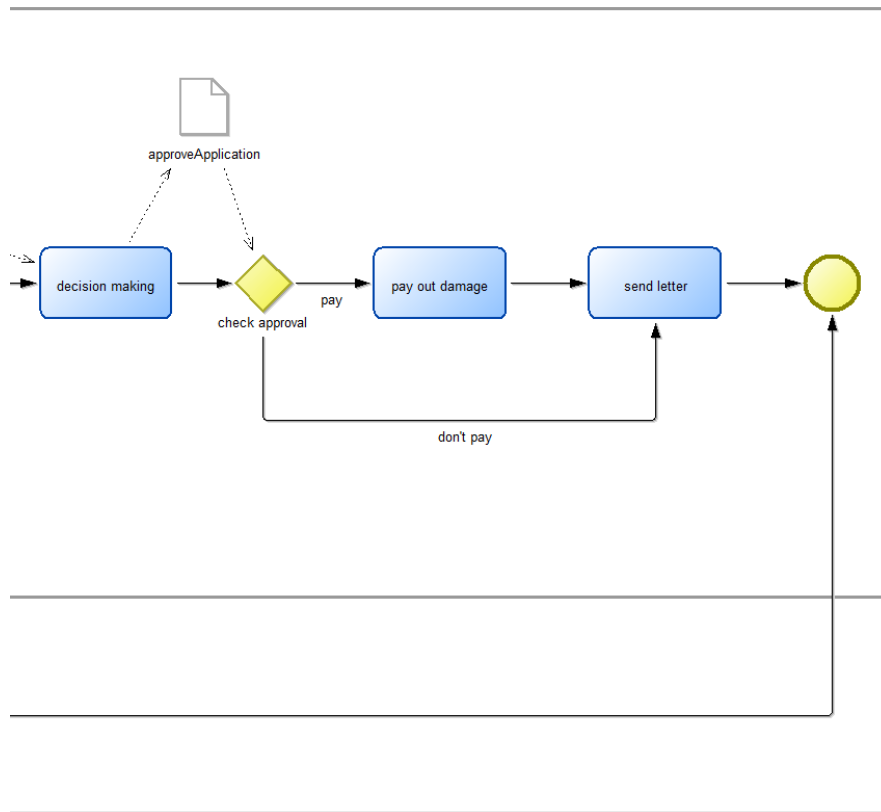


Figure 66: Shows the process view of a visitor with the changes from the original process to this view. To show the changes better, they are marked with colour.

## Merged model

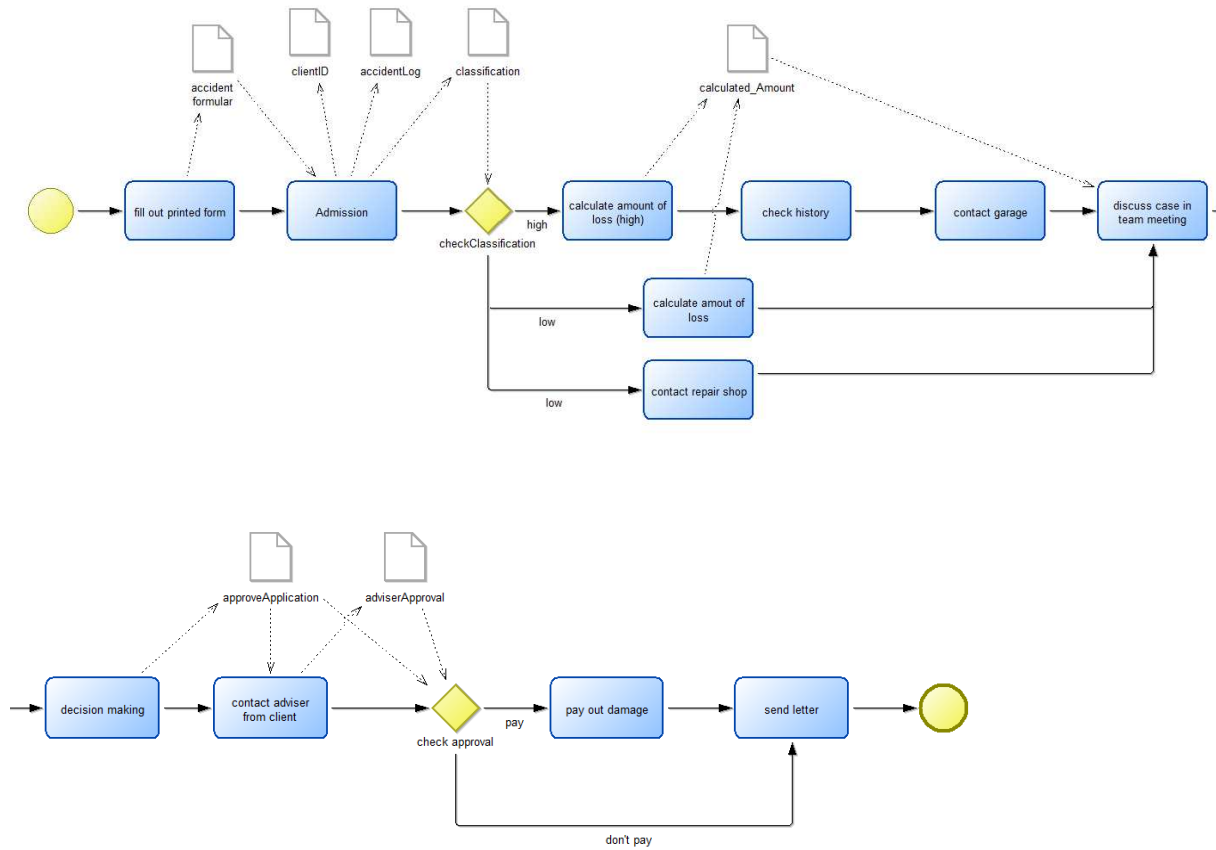


Figure 67: Shows the developed merged model of test case three the process for a KFZ insurance case.

## Merged model colour

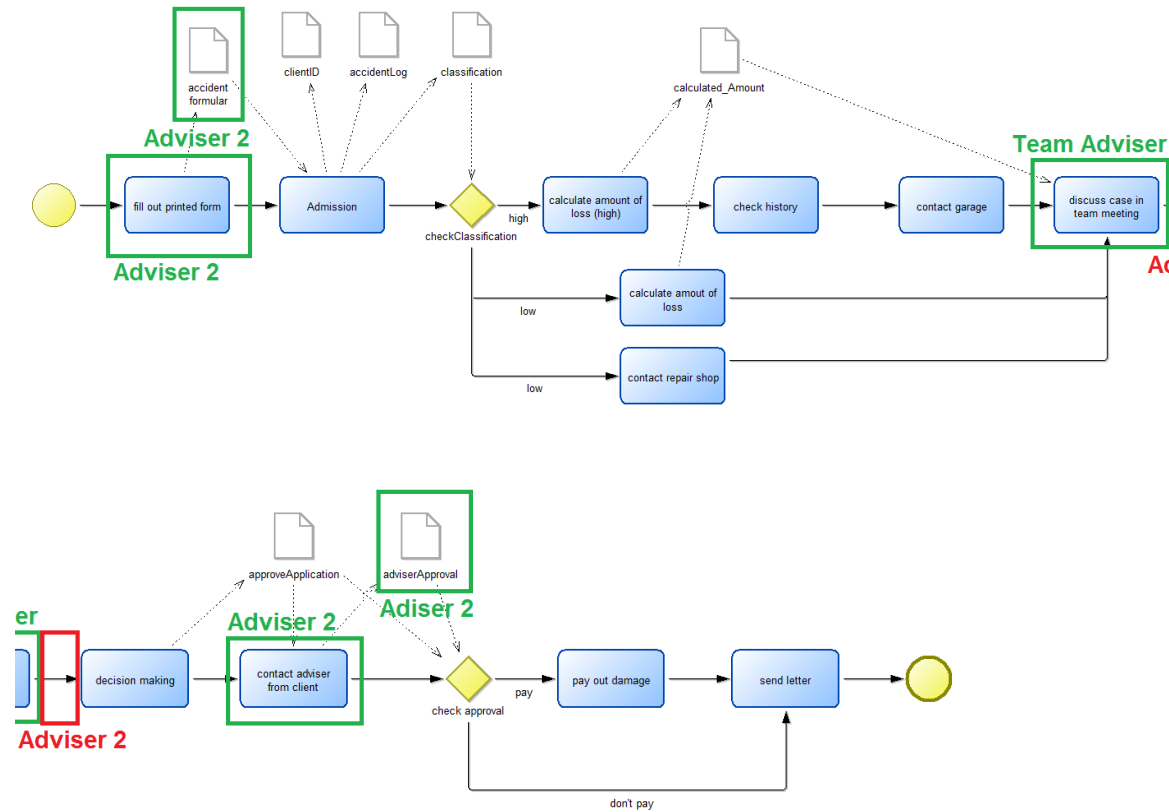


Figure 68: Shows the merged model from test process KFZ with the marked changes to the base view. The changes are marked with different colours and to show from which view the change comes the name of the view is written over or below the structural element.

## 12.5 Abstract

### 12.5.1 English

Most companies already use modelling to document the internal structure. The first recording is rather cost- and time-consuming which leads to the point that the generated models aren't maintained. Most of the work is still manual labour because there aren't any automated ways of gathering processes. The gathering of the relevant information for the models is mostly executed by interviewing the relevant people. The method isn't in every case a real interview when different methods such as a questionnaire are used. After gathering the needed information they must be analysed and then be put together in a model. This master thesis begins at the point where the relevant information is known and introduces a method on how to get a merged model automatically.

### 12.5.2 German

Modellierung ist bereits in die meisten Unternehmen integriert, jedoch ist das erstmalige Erfassen bereits so Zeit- und Kostenaufwändig, das die meisten Unternehmen keine Wartung ihrer Prozesse vornehmen. Weil es noch keine automatisierten Vorgänge beim Erfassen von Prozessen gibt, ist diese Arbeit immer noch ein hauptsächlich manueller Prozess. Um die relevanten Informationen für Prozesse überhaupt zu gewinnen, müssen in irgendeiner Weise, z.B. Fragebögen, die ausführenden Personen befragt werden. Diese gesammelten Informationen müssen im Anschluss ausgewertet und in ein Modell zusammengefasst werden. Diese Arbeit setzt an der Stelle an, wo die individuellen Informationen vorliegen und stellt eine Möglichkeit vor, wie ein Modell automatisch generiert werden kann.

## 12.6 Zusammenfassung

Das Ziel dieser Masterarbeit war es, einen neuen Weg zu finden um verschiedene Sichten von Personen auf ein und denselben Prozess miteinander zu verknüpfen. Nach dem die erste Idee für diese Umsetzung weiter ausgebaut wurde, wurde eine Strategie entwickelt, wie die Umsetzung dieser Aufgabe ablaufen soll. Diese Strategie wurde in einem Model grafisch festgehalten. Ein großer Teil der Arbeit war darauf fixiert, dass geeignete Faktoren gefunden werden mussten, die solch ein Zusammenfassen erlaubten. Dafür wurden zuerst potenzielle Faktoren aus der Modellstruktur und dem vorher fixierten Programm, das zur Modellierung verwendet wurde, herausgefiltert. Nach der Festlegung von zu erfüllenden Kriterien wurden die gefundenen Faktoren bewertet und auf ein paar wenige reduziert. Nach dieser Analyse wurde eine Logik entwickelt die das Zusammenfügen ausführte. In dieser Arbeit finden sich die einzelnen Schritte wie dieser Algorithmus entwickelt wurde. Nach der Entwicklung musste die Logik getestet werden. Zu diesem Zweck wurden drei verschiedene Prozesse herangezogen, welche in ein Abteilungsszenario eingepflegt wurden. Mit Abteilungsszenario ist gemeint, dass eine entsprechende organisatorische Hierarchie festgelegt wurde, um die Variationen des Prozesses zu erklären. Da alle drei Beispiele aus unterschiedlichen Bereichen kommen, sind auch die Szenarien unterschiedlich. Der erste Testprozess kommt aus der Bankenwelt und ist von einem realen Unternehmen zur Verfügung gestellt worden. Die anderen zwei Prozesse befassen sich mit einem Benotungsprozess einer Lehrveranstaltung an der Universität und einem KFZ Versicherungsfall. Diese beiden Beispielprozesse wurden aus einem Aufgabenblatt der Lehrveranstaltung Workflow Technologies an der Universität Wien entnommen. Nachdem die Szenarien mit den Varianten des Prozesses definiert wurden, wurde der Algorithmus auf sie angewandt. Die Ergebnisse wurden zwei Mal analysiert um als Ergebnis aussagen zu können, wie nahe das generierte Modell dem Originalmodell kommt. Die noch offenen Punkte zu diesem neuen Ansatz wurden anschließend diskutiert und in einem Kapitel zukünftige Arbeiten zusammengefasst.

## 12.7 Curriculum Vitae

### Personal information

---

Name: Claudia Gratz

### Work experience

---

06/07.2011	Raiffeisenlandesbank NÖ-Wien Summer Traineeship in department Treasury Mid Office, Support in the development of automated reporting, support in software development projects
10.2011 – 05.2012	Raiffeisenlandesbank NÖ-Wien Marginally employed person, support in software development projects with the programming language SAS Treasury Mid Office / Risk Office Finance
07/2012	Raiffeisenlandesbank NÖ-Wien Trainee for one month, completing a SAS software project from the last employment Treasury Mid Office / Risk Office Finance
10.2012 – 12.2013	Raiffeisenlandesbank NÖ-Wien Marginally employed person, support of optimisation from programmed SAS routines Market Risk Management

### Education

---

October 2009 – July 2012	bachelor studies Wirtschaftsinformatik, University of Vienna
Since October 2012	master studies Wirtschaftsinformatik, University of Vienna