Analysis of the Volvo IT Incident and Problem Handling Processes using Process Mining and Social Network Analysis

Zbigniew Paszkiewicz and Willy Picard

Department of Information Technology, Poznań University of Economics,
al. Niepodległości 10, 61-875 Poznań, Poland
{zpasz,picard}@kti.ue.poznan.pl

Abstract. Volvo IT Belgium has provided real-life datasets for the need of the Third International Business Process Intelligence Challenge, in the form of event logs generated by the VINST system used across the Volvo corporation to support incident and problem handling. Volvo has also pointed 4 aspects of their business operations they would like to be scrutinized. 1) Are the incidents contained within the 1st support line? 2) Is bouncing delegation, a.k.a. ping-pong, a frequent phenomenon? 3) Are employees cheating on VINST by faking inexistent waits from end-users? 4) Are real process instances conform with the process model proposed by Volvo IT? In this paper, we provide answers to all these questions using process mining and social network analysis techniques, and we state the existence of hidden support lines degrading the overall performance of incident handling, little localized ping-pong behavior and wait-user misuse, and various levels of conformity across organizations.

Key words: business process intelligence, process mining, social network analysis, incident management, problem management, Volvo

1 Introduction

Volvo IT provides IT services according to terms and conditions regulated in Service Level Agreements (SLAs). The VINST system is used by Volvo IT to support incidents and problems reported by the IT service users. The primary goal of the *Incident Handling* process is to “restore normal service operation as quickly as possible and by that ensure that the best possible levels of service quality and availability are maintained” [1]. The *Problem Handling* process includes “all the activities required to diagnose the root cause(s) of incidents and to secure the resolution of those problems to enhance the quality of IT-services delivered and/or operated by Volvo IT” [1]. In this paper, incidents and problems are commonly referred to as *requests*.

Guidelines for the execution of both processes follow from the ITIL approach [2]. In particular, processes are organized around three support lines. Each *incident* is preferred to be solved by Support Teams (STs) working in the

1st line. The incident is forwarded to the other lines only when it cannot be handled in the 1st line. Meanwhile, *problem* handling is primary the responsibility of the 2nd and the 3rd line STs. STs operation is spread across organizations and function divisions in Volvo IT.

The data analyzed in this paper come from the VINST system. Volvo IT has pointed the following aspects of request handling as particularly interesting and has asked for in-depth analysis:

1. *push to front strategy*: this aspect refers to the Volvo IT policy encouraging incident handling mainly in the 1st line STs; every time an incident is forwarded to the 2nd and the 3rd line, it has a negative impact on the process efficiency and the efficiency of the Volvo IT as a whole;
2. *ping-pong behavior*: this aspect refers to the unwanted pattern in interaction among STs, when a request is repeatedly bounced from one ST to another;
3. *wait user sub-status use*: this aspect refers to the unwanted mis-use of the "wait-user" activity sub-status available in the VINST system; the overuse of this status indicates the hidden inefficiency of the process;
4. *process conformity per organization*: this aspect refers to the compliance of the real process instances with an "ideal" designed process model.

Specific questions were asked by Volvo IT concerning all these four aspects [1]. The Incident Handling process is the center of attention throughout the analysis. The Problem Handling process is addressed only by the second aspect, i.e., ping-pong behavior.

In this paper, we present the results of our analysis of the data provided by Volvo IT, with the help of both process mining [3] and social network analysis techniques [4]. In Section 2, an overview of the datasets is given. Our answers to the proposed questions are detailed in Section 3. Next, the tools used for the analyses are presented in Section 4. Finally, Section 5 concludes the paper.

2 Understanding the process

The provided event log is a record of activities undertaken by Volvo IT employees during the execution of Incident Handling [5] and Problem Handling [6, 7] processes. Three files have been provided: (1) an event log concerning Incident Handling process, (2) an event log concerning Problem Handling process for still opened problems, (3) an event log concerning Problem Handling process for closed problems. Each event of the logs corresponds to an employee activity, where an activity is a change in request status and/or sub-status.

Only slight modifications of the event log were required in order to start the analysis: first, data of the "support team" column have been divided into two columns ⟨"support team", "support line number"⟩, with the values of "support line number" being in the set{"1st", "2nd", "3rd", "2nd-3rd"}; second, the 1st line number was assumed for activities for which no line number was mentioned; third, the format of timestamps has been modified to enable their processing by various tools.

During data munging, the following inconsistencies within the datasets has been identified: first, some statuses and sub-statuses of activities not specified in the dataset [1] are present in the datasets: e.g., wait-customer, in-call, unmatched, resolved (all the statuses and sub-statuses are taken into account in our analyses); second, for some activities, no organization or function divisions are indicated (activities without those values were left out in our analyses); names and symbols of countries in columns `Country` and `Owner Country` in some cases are sometimes inconsistent (the `Country` column was used in our analyses).

2.1 The Incident Handling event log

The Incident Handling event log captures 65.533 events generated during the execution of 7.554 process instances. The events were recorded in a time period from 31.10.2010, 15:59:42 to 23.05.2012, 00:22:25. 81% of all the events have been recorded from April, 16th to May, 19th 2012. Activities are associated with 705 different products. Products associated with the highest numbers of process instances are products `424` (882 process instances, 11.6%), `660` (484, 6%), `253` (226, 2.9%), `383` (205, 5%).

There are 13 activity types in the Incident Handling log: "Accepted/In Progress" (46.14% of all the activities), "Queued/Awaiting Assignment" (17.62%), "Completed/Resolved" (9.33%), "Completed/Closed" (8.72%), "Accepted/Wait - User" (6.43%), "Accepted/Assigned" (4.92%), "Completed/In Call" (3.11%), "Accepted/Wait" (2.34%), "Accepted/Wait - Implementation" (0.75%), "Accepted/Wait - Vendor" (0.48%), "Accepted/Wait - Customer" (0.15%), "Unmatched/Unmatched" (0.01%), "Completed/Cancelled" (0.001%).

The majority of Handle Incident process instances starts with an "Accept/In Progress" (84.35%) or "Queued/Awaiting Assignment" (15.3%) activity. Four activities never start the Incident Handling process: "Assigned/Wait-customer", "Unmatched/Unmatched", "Completed/Closed", "Completed/Cancelled". The majority of process instances is finished by "Completed/Closed" (73.77%) while process instances finished by "Completed/In Call", "Completed/Resolved" and "Completed/Cancelled" account for 26.1% of all the process instances. Eight process instances (0.1%) is still running.

The most frequent behavior is presented in Fig 1. Only the most frequently executed activities and transitions are visible. Numbers assigned to activities and transitions indicate the number of process instances they appeared in.

There are 2.278 variants of Incident Handling process execution. The most frequent variant (1.749 executions, 23.15%) consists of three steps: "Accepted/In Progress", "Accepted/In Progress", and "Completed/In Call". The second most popular process variant (524 executions, 6.94%) consists of four steps: "Accepted/In Progress", "Accepted/In Progress", "Completed/Resolved", "Completed/Closed". Although the number of events varies among variants from 1 to 123, 14% of process instances are performed in less than 10 minutes. The maximal duration of a process instance is 2 years 41 days.

The Incident Handling process is rather unstructured, as illustrated in Fig. 2. All the activities recorded from April, 16th to May, 19th 2012 are presented in

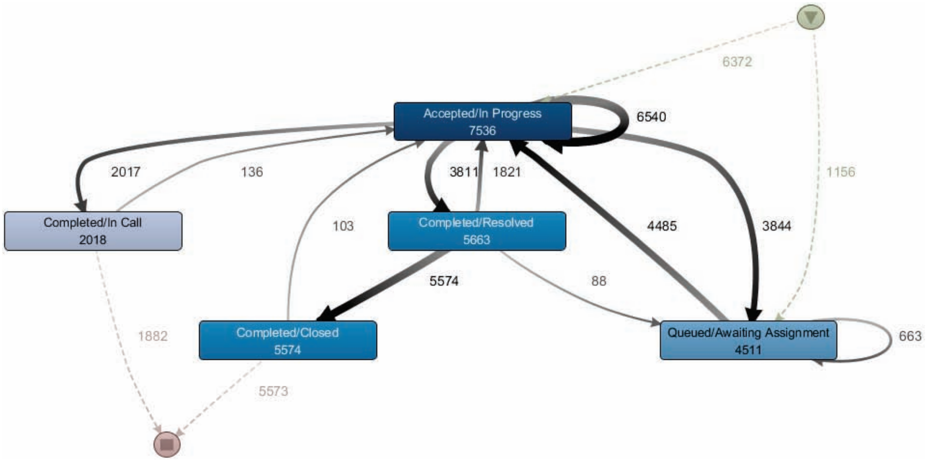


Fig. 1. Typical Handle Instance process execution.

Fig. 2a). The wide variety of transitions among activities leads to *spaghetti-like diagram*. The Incident Handling process is presented in Fig. 2b)as a dotted chart. Each row corresponds to exactly one process instance and each dot corresponds to an activity. Process instances are sorted from the top according to start time. The color of the dot is associated with activity type. One may notice an increasing number of requests reported over time. The wide variety of the process instances duration, another typical characteristic of weakly structured spaghetti-like processes, is clearly visible. Finally, the execution of "Queued/Awaiting Assignment" (represented by blue dots) and "Completed/Closed" (represented gold dots) activities is synchronized across many process instances.

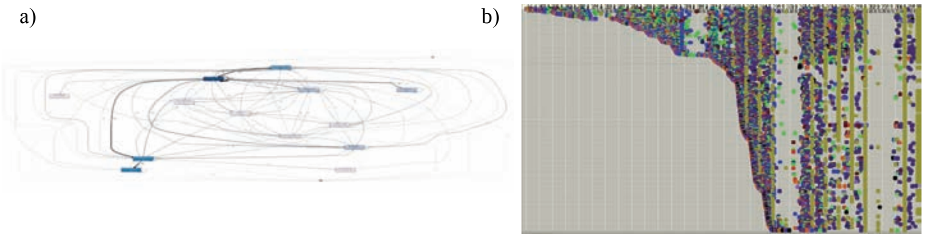


Fig. 2. Unstructured character of Incident Handling process.

According to the event log, Volvo IT is divided into 24 organizations, 594 STs, and 605 function divisions. Three organizations are especially active: C (41.189 activities, 64.38%), A2 (12.508, 19.09%), and B (4.623, 7.05%). Four STs execute almost 30% of work: G97 (7.466 activities, 11.39%), G96 (5.999, 9.15%), S42 (4.382, 6.69%), and G230 (1.661, 2.53%). The three most active function divisions are V3_2 (30.950, 47.23%), A2_1 (9.977, 15.22%), and E_10 (4.527, 9.41%).

The geographical distribution of activities is presented in Fig. 3. 46.042 (70.26%) activities are performed in the 1st line, mainly in Sweden, Poland, Brazil, and India. The most active countries in the 2nd line are Sweden, India, Poland, and Brazil, performing 16.541 (25.24%) activities. 2.911 activities (4.44%) are performed in the 3rd line, mainly in Sweden, Poland, France, and India. Finally, "2nd 3rd" support line activities are handled in France, performing 39 activities (0.06%).

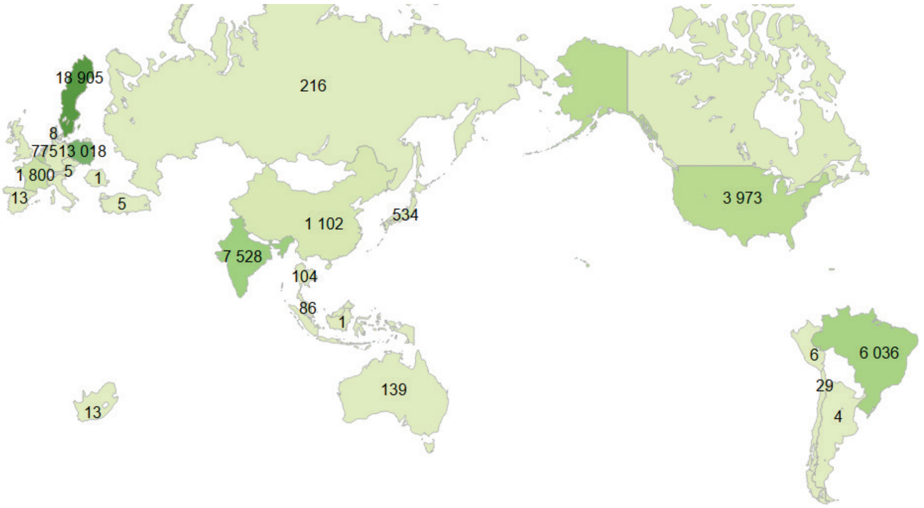


Fig. 3. Geographical distribution of activities. Numbers indicate the volume of activities performed in each country.

2.2 The Problem Handling event logs

The Problem Handling process is much more structured in comparison with the Instance Handling process, although it still cannot be called *lasagna-like* process. In Fig. 4a), the full behavior recorded in the closed event log is presented in a form of the control-flow diagram. Although the number of performed activities and the number of possible transitions are significantly smaller in the Problem Handling process than in the Incident Handling process, the dotted chart presented in Fig. 4b) illustrates the still unstructured character of the Problem Handling process.

A comparison of the Opened and Closed Problem Handling processes is presented in Tab. 1. The typical behavior for both logs is presented in Fig. 5.

In Fig. 5, the numbers associated with activities or transitions indicate the number of process instances passing through a particular activity or transition.

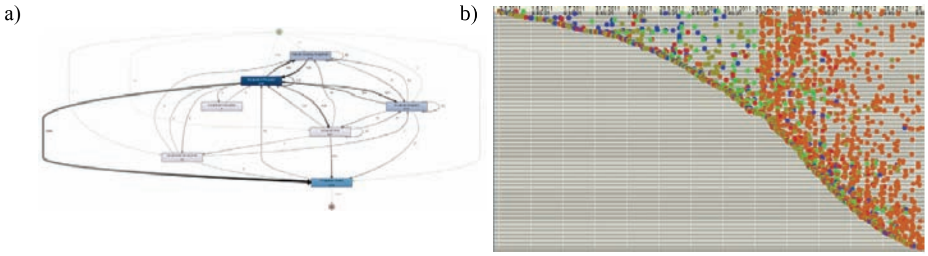


Fig. 4. Level of structuration of the Closed Problem Handling process.

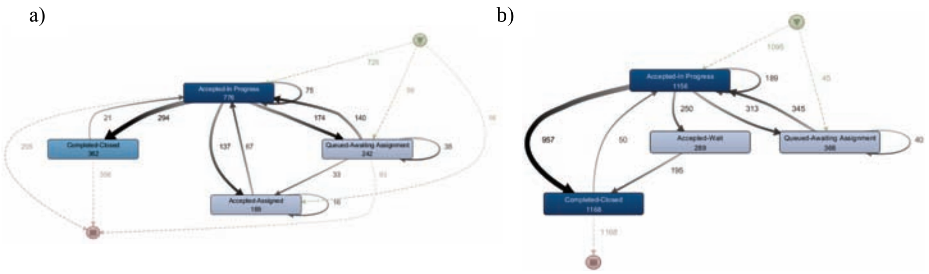


Fig. 5. Typical behavior of (a) opened problems, (b) closed problems.

The general conclusion is that, although the size of the logs is different, registered typical behaviors are similar in both Opened and Closed Problem Handling processes.

3 Questions

The four aspects have been investigated during the event log analysis: *push to front strategy*, *ping-pong behavior*, *wait-user sub-status use*, and *process conformity per organization*.

3.1 Push to Front Strategy

In general, the Incident Handling process follows the push to front strategy. The most frequent variants of the Incident Handling process are presented in Fig. 6. As a general rule, incident handling is performed mostly in the 1st line. The number of process instances passed to 2nd and 3rd line is smaller by at a least an order of magnitude.

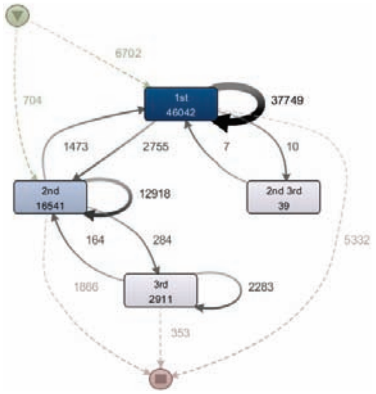


Fig. 6. Typical incident handling.

Table 1. Comparison of the Open and Closed Problem event logs

Statistic	Open Problems	Closed Problems
Events	2351	4755
Process instances	819	1166
No. variants	183	236
Typical variant	Accepted/In Progress, Completed/Closed (21.49%, mean duration: 49 days, 1 hours)	Accepted/In Progress, Completed/Closed (38.81%, mean duration: 49 days, 6 hours)
Second variant	Accepted/In Progress (15.02%, mean duration: 0)	Accepted/In Progress, Accepted/Wait, Completed/Closed (7.45%, mean duration: 80 days, 19 hours)
Activities	Accepted/In Progress (49.09%) Completed/Closed (16.46%) Queued/Awaiting Assignment (16.29%) Accepted/Wait (9.1%) Accepted/Assigned (9.02%) Assigned-Assigned (0.04%)	Accepted/In Progress (46%) Completed/Closed (25.97%) Queued/Awaiting Assignment (12.95%) Accepted/Wait (7.55%) Accepted/Assigned (7.11%) Completed/Cancelled (0.06%)
Minimal duration	0 millisecond	1minute 22 seconds
Maximal duration	4 years 300 days	356 days 22 hours
Most active support lines	2nd (68.14%), 3rd (28.92%), 1st (2.93%)	2nd (64.27%), 3rd (30.52%)
Most active support teams	G42 (7.36%), S33 (7.19%)	G199 (9.86%), G88 (5.89%)
Most active organizations	C (47.89%), A2 (26.03%), G4 (14.12%)	C (50.58%), A2 (24.46%), G3 (9.86%)
Most function division	E_10 (14.12%), A2_1 (12.29%)	C_6 (8.71%), E_10 (8.56%)

To understand the exceptions to the push to front strategy, we have performed a two-step analysis. First, exceptions to the push to front strategy are studied from the product perspective, i.e., products that follow and do not follow the strategy are indicated. Second, Volvo IT organizational structures are checked with regard to their respect of the push to front strategy.

Products. All the products were analyzed in terms of involvement of support lines with two approaches: (1) analysis of transitions among support lines, (2) analysis of number of activities per lines.

In the first approach, 12 methods of incident handling were distinguished depending on the set and ordering of involved support lines. Correct methods, i.e., methods supporting the push to front strategy, include: process instances fully performed in the 1st line (marked as "1"), process instances started in the

1st and forwarded to the 2nd line (1-2), process instances started in the 1st and forwarded to the 2nd and later to the 3rd line (1-2-3). Incorrect methods are: 2, 3, 1-3, 2-1, 2-3, 2-1-3, 3-1, 3-2, 3-1-2.

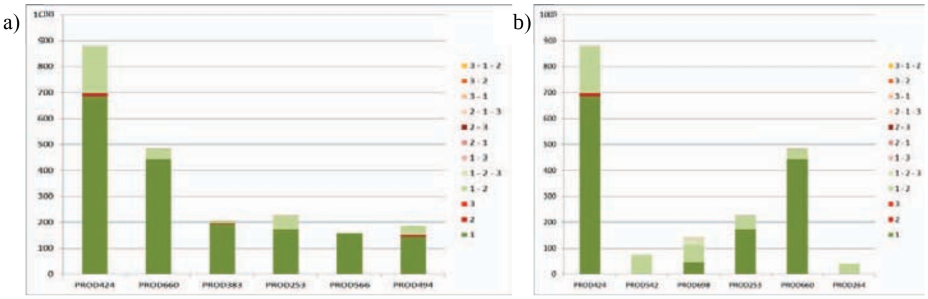


Fig. 7. Products well aligned with push to front strategy: a) products following the "1" method; b) products following the "1-2" method.

In Fig. 7, products that are well aligned with push-to-front strategy are presented, including products for which most of the process instances are fully executed in the 1st line, such as 424, 660, 383, 253, 566, and 494 (cf. Fig. 7a)).

The two most frequently executed products (424 and 660) are worth a closer look. 7.364 and 2.728 activities are performed in the 1st line for products 424 and 660 respectively. Handling incidents for these set of products include also methods "1-2" and "2". In case of these products, the incorrect method "2 is performed very rarely.

In Fig. 7b), products are ordered according to the number of process instances following the "1-2" method. Products are still well aligned with push-to-front strategy but often forwarded to the second line. This set includes products 424, 542, 698, 253, 660, and 264.

Products frequently requiring the support of the 3rd line, e.g, product 698, are presented in Fig. 8. It is noticeable that for most products in this set, e.g., 604, 295, 617, 611, the incorrect-to-correct-executions ratio is important, with a significant presence of the "2", "3", and "1-3" methods.

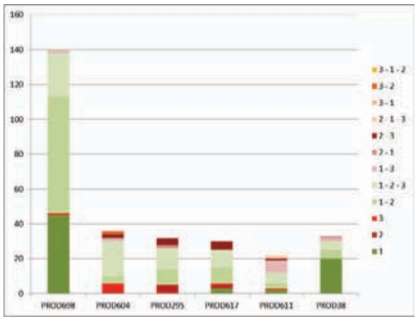


Fig. 8. Products following the "1-2-3" method.

In Fig 9, the set of 30 most frequently supported products (corresponding to 53% of all the activities) is presented (without products 424 and 660 discussed earlier).

Table 2. Number of activities performed for 10 most popular products by support line

	Products									
Support Line	13	253	267	321	383	453	494	544	566	698
1st	713	1608	769	640	839	325	1208	319	651	877
2nd	132	261	229	99	49	109	213	201	-	424
3rd	19	-	3	-	2	3	-	-	-	193

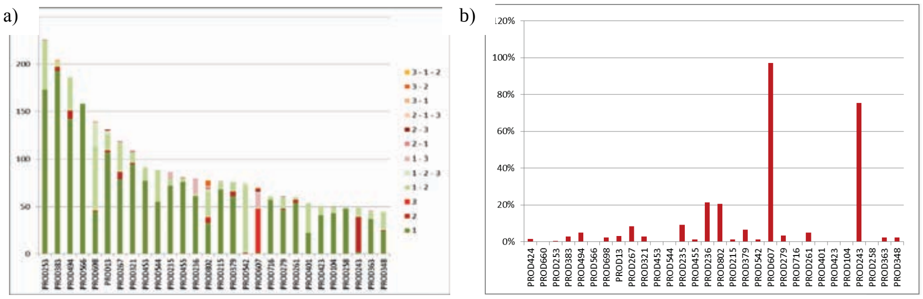


Fig. 9. The 30 most frequently supported products. On the left side, the execution methods for each selected product. On the right side, the percentage of wrong executions for each selected product.

The majority of products presented in Fig. 9a) are well aligned with the push to front strategy. This claim is supported also by Tab. 2 in which the number of activities per support line and per product is presented for the 10 most supported products from Fig. 9. The majority of activities are performed in the 1st line.

In Fig. 9a), two significant exceptions—products 607 and 243—do not follow the push to front strategy. This observation is confirmed by Fig. 9b). In Fig. 9b), the height of bars indicates the percentage of wrong executions. The percentage is especially high for the formerly mentioned products 607 and 243 as well as products 235, 236, and 802. The most frequent behavior for product 607 is illustrated in Fig. 10: all the activities during the incident handling are performed in the 3rd line.

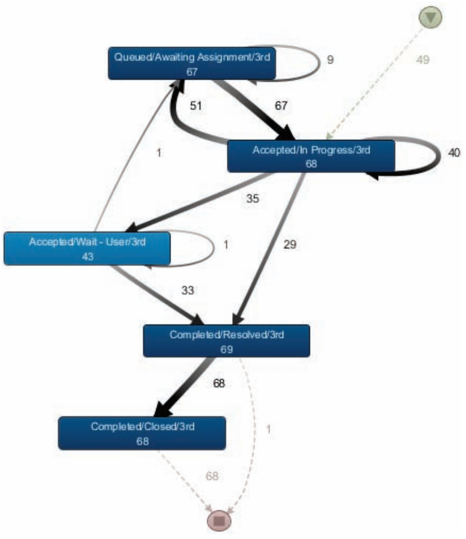
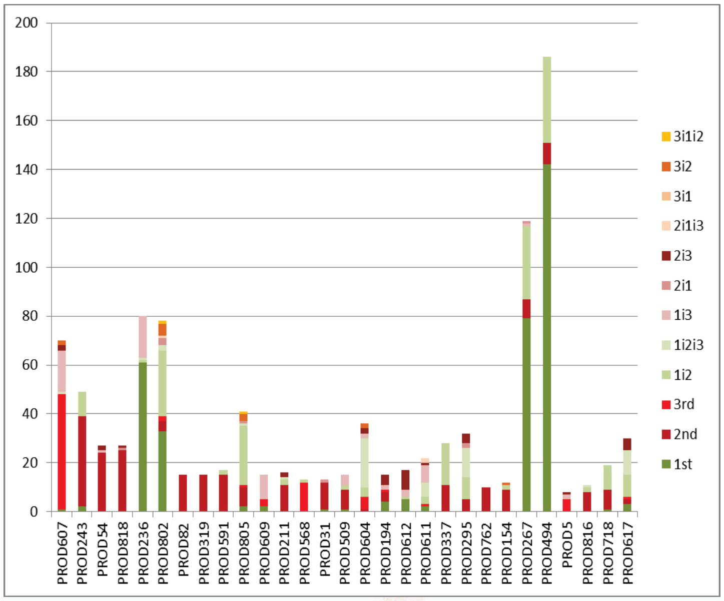
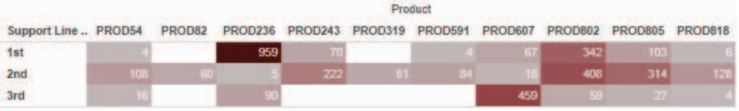


Fig. 10. Common behavior in incidents handling for product 607.

Products following incorrect execution methods are presented in Figs. 11 and 12. In Fig. 11, products are ordered according to the absolute number of wrong executions (without product 424 discussed earlier). The sets of products presented in the two figures are similar. Products that do not follow the push to front strategy are products 607, 243, 54, 818, 82, and 319.





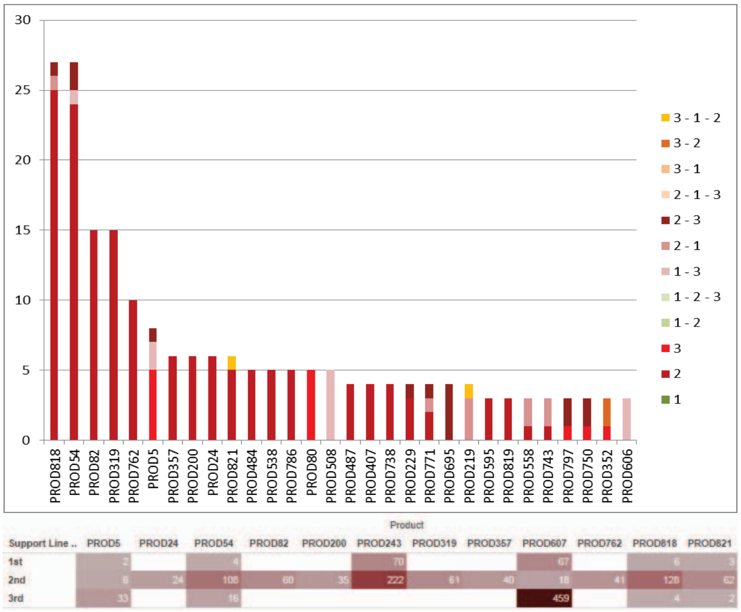
	Product									
Support Line ..	PROD54	PROD82	PROD236	PROD243	PROD319	PROD591	PROD607	PROD802	PROD805	PROD818
1st	4		959	70		4	67	342	103	6
2nd	108	60	5	222	61	84	18	408	314	128
3rd	16		90				459	59	27	4

Fig. 11. Products with the highest absolute number of incorrect process instance executions.

The typical behavior of products 818 and 54, exceptions to the push to front strategy, is presented in Fig. 13: all the activities are performed in the 2nd and the 3rd line. The numbers associated with activities and transitions indicate the numbers of process instances the activity and transitions were performed in.

Products most frequently pushed to the 2nd line and fully executed in this line are products 243, 818, 54, 591, 82 and 319 (Fig. 14a). Products frequently handled in the 3rd line are 607, 568, 604, 5, 80, and 609 (Fig. 14b).

131 products do not appear in the event log often but are systematically handled in an incorrect way, with 322 such incorrect executions. Among these 131 products, 9 are always handled only by the 3rd line (80, 70, 74, 102, 578, 89, 75, 57, and 610), and 71 products are handled only in the 2nd line (e.g., 82, 319, and 762).



Support Line ..	PROD5	PROD24	PROD54	PROD82	PROD200	PROD243	PROD319	PROD357	PROD607	PROD762	PROD818	PROD821
1st	2		4			70			67		6	3
2nd	6	24	108	60	35	222	61	40	18	41	128	62
3rd	33		16						459		4	2

Fig. 12. Products with the highest percentage of incorrect process instance executions.

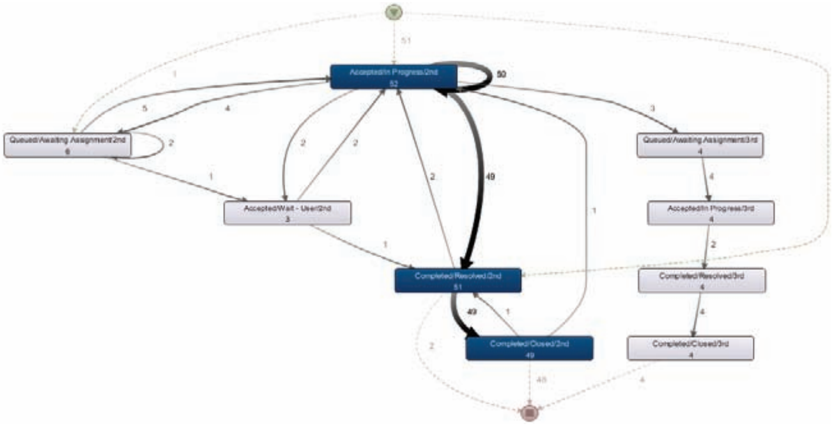
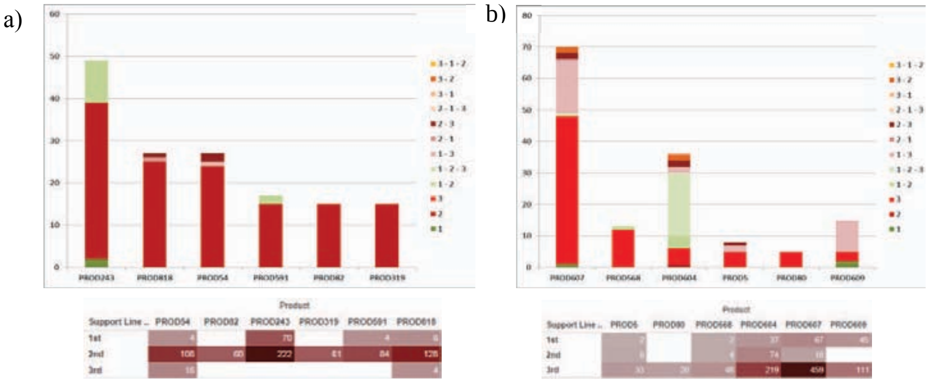


Fig. 13. Typical behavior in incident handling for products 818 and 54.

Volvo IT organizational structures. A first hypothesis is that functions are attached to lines. If this hypothesis were correct, a given function would be always performed on a given line. Then, lines would be considered as separated by functions.

However, the analysis of the incident dataset leads to the disproval of this hypothesis: *functions are mainly scattered among lines*, as presented on the chord diagram in Fig. 15.



	Product					
Support Line ...	PROD54	PROD82	PROD243	PROD319	PROD591	PROD818
1st	4		70		4	6
2nd	106	65	222	61	84	128
3rd	16					4

	Product					
Support Line ...	PROD5	PROD80	PROD568	PROD604	PROD607	PROD609
1st	2		2	37	67	45
2nd	6		4	74	18	
3rd	33	20	48	219	459	111

Fig. 14. Products with incidents pushed and solved in: a) 2nd line, b) 3rd line.

In a general manner, chord diagrams shows relationships among groups of entities. Each block on the exterior circle represents an entity. The links between entities represent shared characteristics of the entities. Additionally, the size of the link area is proportional to the number of common characteristics of the entities. The ticks on the exterior side of the circle aim at providing guides to the number of common characteristics.

On this diagram, four lines “1st line”, “2nd line”, “3rd line”, and “2nd-3rd”, matching the line names provided in the provided data sets, are represented by 4 separated blocks on the outside circle. Two blocks are linked according to the number of functions that they both perform. A particular case is the case of functions performed by a unique line. In this case, an area attached to a single block is drawn.

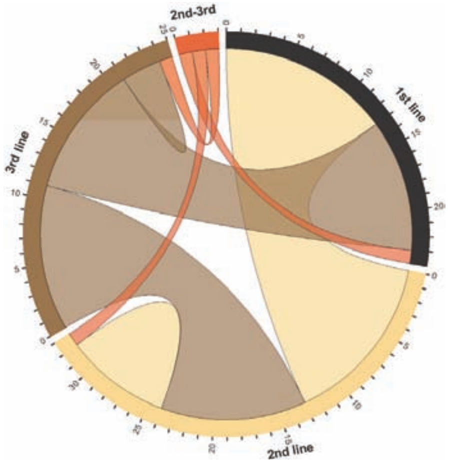


Fig. 15. Function sharing across lines.

From the chord diagram, one may identified that no function performed by the 1st line is performed only by the 1st line. 13 functions performed by the 1st line are performed by the 2nd line too, 10 functions are performed by the 3rd line. 11 functions are performed both by the 2nd and 3rd lines. The “2nd-3rd” line performs 1 function in common with the 1st line, another function with the 2nd line, and a last function with the 3rd line. Only 8 functions are performed only by the 2nd line, and 3 functions are performed only by the 3rd line. A summary of the number of functions performed by two lines is presented in Tab. 3.

Table 3. Functions co-performed by lines

	1st line	2nd line	3rd line	2nd-3rd line
1st line	0	13	10	1
2nd line	13	8	11	1
3rd line	10	11	3	1
2nd-3rd line	1	1	1	0

Another hypothesis is that organization lines are spread amongst lines, assuming that the competences are scattered within each organization line among all the lines. If this hypothesis were correct, each organization line would be operating on each line and no organization line would be confined to a single line.

The analysis of the incident dataset leads to a partial confirmation of this hypothesis: *organization lines are mainly participating in many lines*, as presented on the chord diagram in Fig. 16. On this diagram, four lines "1st line", "2nd line", "3rd line", and "2nd-3rd", matching the line names provided in the provided data sets, are represented by 4 separated blocks on the outside circle. Two blocks are linked according to the number of organization lines participating in these lines. A particular case is the case of organization lines performing on a single line. In this case, an area attached to a single block is drawn.

From the chord diagram, on may identify that 9 organization lines are active on both the 1st and the 2nd lines, 5 organization lines are active on both the 1st and the 3rd lines, and 7 organization lines are active on both the 2nd and the 3rd lines. One organization line active on the "2nd-3rd" line is also active on the 1st line, another one on the 2nd line, and a last one on the third line. Therefore, organization lines tend to participate in more than one line.

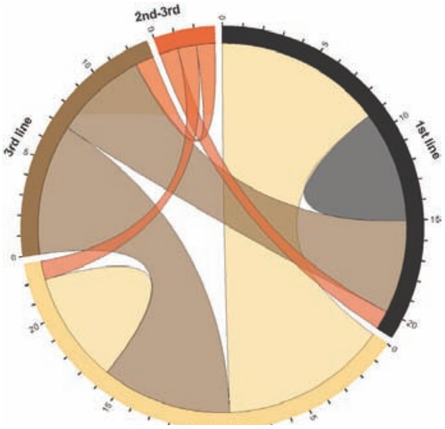


Fig. 16. Organization lines across lines.

However, 6 organization lines operate only on the 1st line and 6 organization lines operate only on the 2nd lines. Therefore, the hypothesis of a systematic spreading of organization lines across lines is not fully acceptable.

A summary of the number of organization lines operating on various lines is presented in Tab. 4.

Another hypothesis is that process owners are attached to a given line. If this hypothesis were correct, a given process owner would performed only on the 1st

Table 4. Organization lines sharing across lines

	1st line	2nd line	3rd line	2nd-3rd line
1st line	0	13	10	1
2nd line	13	8	11	1
3rd line	10	11	3	1
2nd-3rd line	1	1	1	0

line, 2nd line, or 3rd line, and no process owner would operate on two or more lines.

The analysis of the incident dataset leads to the disproval of this hypothesis: *a large number of process owners is operating on many lines*, as presented on the chord diagram in Fig. 17. On this diagram, four lines "1st line", "2nd line", "3rd line", and "2nd-3rd", matching the line names provided in the provided data sets, are represented by 4 separated blocks on the outside circle. Two blocks are linked according to the number of process owners operating on these lines. A particular case is the case of process owners operating on a single line: an area attached to a single block is then drawn.

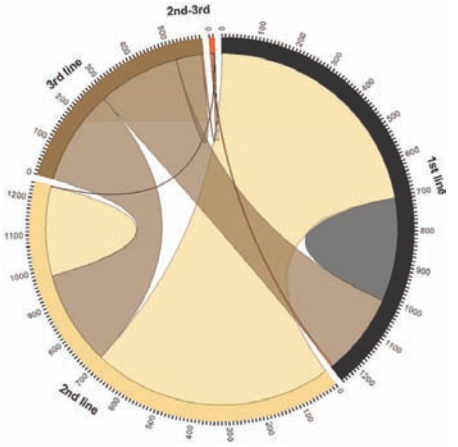


Fig. 17. Process owner involvement in lines.

About 700 process owners are operating on both the 1st and 2nd lines. About 250 process owners are operating on both the 1st and the 3rd lines. Almost 300 process owners are operating on both the 2nd and the 3rd lines. The process owners operating on the "2nd-3rd" line are operating on the 1st line (5 process owners), 2nd line (2 process owners), and the 3rd line (5 process owners). Therefore, most process owners are operating on two lines or more.

However, still a large number of process owners are operating on a single line: about 300 process owners are working only on the 1st line, about 250 process owners are working only on the 2nd line, and less than 100 process owners are working only on the 3rd line. Therefore, a core of process owners are focusing only on a single line while a large number of process owners are operating in a cross-line mode.

A summary of the number of process owners operating on various lines is presented in Tab. 5.

Another hypothesis is that STs are attached to a given line. If this hypothesis were correct, a given ST would operate only on a given line, and no ST would operate on two lines.

Table 5. Process owners by lines

	1st line	2nd line	3rd line	2nd-3rd line
1st line	303	704	239	5
2nd line	704	261	276	2
3rd line	239	276	75	5
2nd-3rd line	5	2	5	1

The analysis of the incident dataset leads to a partial confirmation of this hypothesis. Indeed, *a large number of STs are operating on a single line. However, a significant number of STs are operating on two lines*, as presented in Fig. 18. On this diagram, four lines "1st line", "2nd line", "3rd line", and "2nd-3rd", matching the line names provided in the provided data sets, are represented by 4 separated blocks on the outside circle. Two blocks are linked according to the number of STs operating on these lines. A particular case is the case of STs operating on a single line: an area attached to a single block is then drawn.

About 209 STs are performing only on the 1st line. About 250 STs are performing only on the 2nd line. About 90 STs are performing only on the 3rd line. Most STs are confided to a given ST. However, about 30 STs are performing on both the 1st and the 2nd lines, while about 15 STs are performing on both the 2nd and the 3rd lines. The unique ST on the "2nd-3rd" line is performing also on the second line. An interesting aspect is the lack of ST operating on both the 1st and the 3rd line. A summary of the number of STs operating on various lines is presented in Tab. 6.

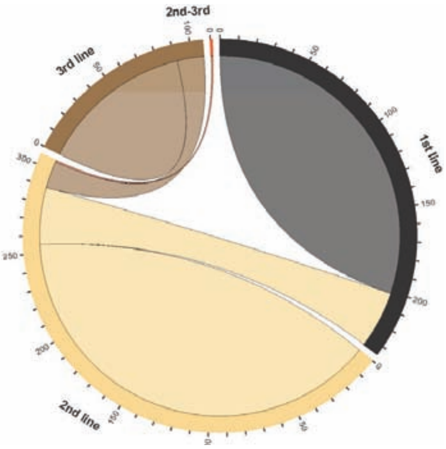


Fig. 18. STs operating in lines.

Therefore, from this analysis, STs are mostly confined to a given line. However, some STs are connecting the 1st line to the 2nd line as well as the 2nd line to the 3rd line, which may be considered as "bridges" gradually linking the lines.

This fact has been investigated further. First, the sets of STs belonging to "bridges" has been identified, leading to our proposal of two additional lines: the "4th line" (1st and 2nd line bridge) and the "5th line (2nd and 3rd line bridge). In Fig. 19, the diagram capturing typical behavior in process instances involving 4th and/or 5th line is presented.

In the process instances in which the "2nd-3rd" line is involved, only the 4th and the 5th lines appear, i.e., the 2nd and 3rd lines are not involved (cf. Fig. 19b). The "2nd-3rd" line interacts only with the 4th line, i.e., receives and

Table 6. STs by lines

	1st line	2nd line	3rd line	2nd-3rd line
1st line	201	34	0	0
2nd line	34	255	16	1
3rd line	0	16	91	0
2nd-3rd line	0	1	0	0

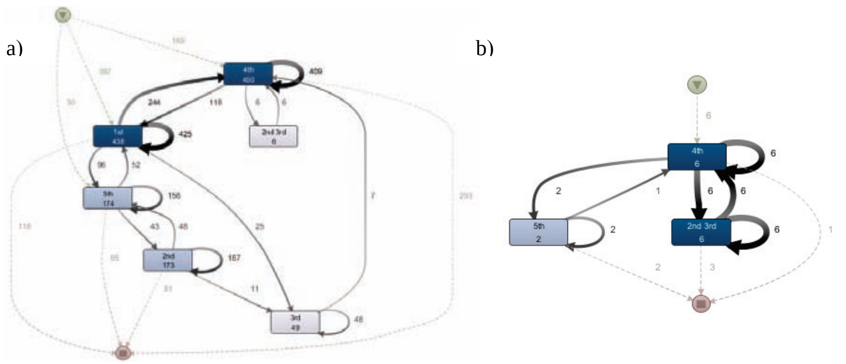


Fig. 19. The 4th and 5th lines: a) most frequent behavior, b) process instances including line 2nd-3rd.

forwards the work to the 4th line (cf. Fig. 19a) and b). Both the 4th and the 5th lines interacts mainly with the 1st and the 2nd line. As a consequence, STs from the 4th and 5th line blur the border between the 1st and the 2nd. Therefore, *the existence of the 4th and 5th lines strongly contributes to difficulties with being in line with the push to front strategy.*

Another hypothesis is that process owners are performing a given function, which is aligned with a function-oriented management. If this hypothesis were correct, a given process owner would focus on a given function and specialization would be the rule for employees.

The analysis of the incident dataset leads to a disapproval of this hypothesis. Indeed, *a very small number of process owners are performing a unique function*, as presented on the chord diagram in Fig. 20. On this diagram, each block on the outside circle represents a function. Two blocks are linked according to the number of process owners performing the associated functions.

For almost each pair of functions, it exists a process owner that performs both functions. The rare exception is function C_5. Process owners performing function C_5 are either performing D_2, E_8, or E_10 (cf. Fig. 20). Therefore function C_5 is either rare, or requires a very narrow of expertise. Another interesting fact is that process owners performing function C_3 are performing most existing functions but C_5, D_2, E_8. Therefore, process owners performing function C_3 seem to "avoid" functions performed by process owners performing C_5.

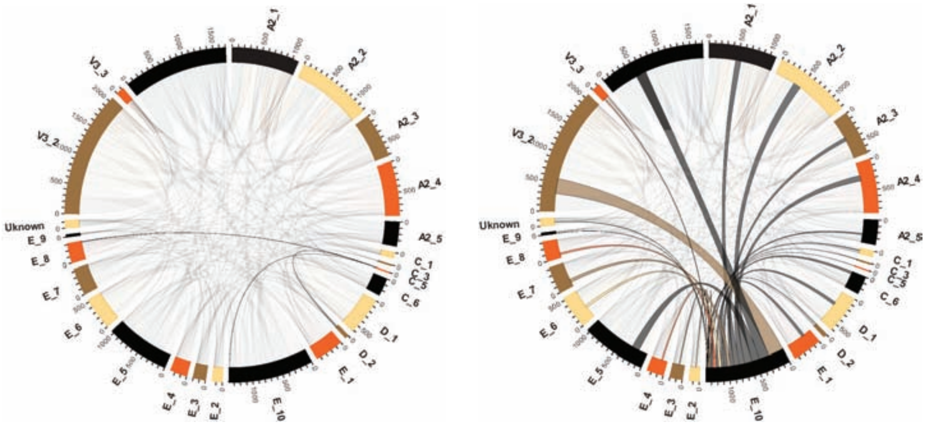


Fig. 20. Functions performed together by employees. On the left side, the set of functions performed by employees performing C_5 are highlighted. On the right side, the set of functions performed by employees performing E_10 are highlighted.

Another hypothesis is that a given organization line is responsible for a given function. If this hypothesis were correct, functions would not be performed by various organization lines.

The analysis of the incident dataset leads to a partial confirmation of this hypothesis. Indeed, *some organization lines tend to share functions with a large number of organization lines, while other organizations do either share their functions with a very limited number of organization lines, or perform functions in an exclusive manner*, as illustrated in Fig. 21.

The first group of organization lines sharing their functions with a large number of STs corresponds to the organization lines B, C, E, F, G1, G4, H, and V* (except V8). These organizations lines share their functions among themselves, which lead to the supposition that this organization lines are either performing a large range of generic operations or that they are very scattered organization lines having a very similar profile. The presence of the large C organization line in this group is to be noted.

The second group of organization lines contains the organization lines A2, D, G2, G3, and V8. These organization lines don't share their functions with a large number of organization lines, but rather either share them with organization lines within this second group. The case of A, D, and V8 is special as these organization lines have specific functions that are performed exclusively by these organization lines.

The case of the organization line C is interesting as on the one hand organization line C share functions with many organization lines, on the second hand, a large set of functions performed by C is only performed by C.

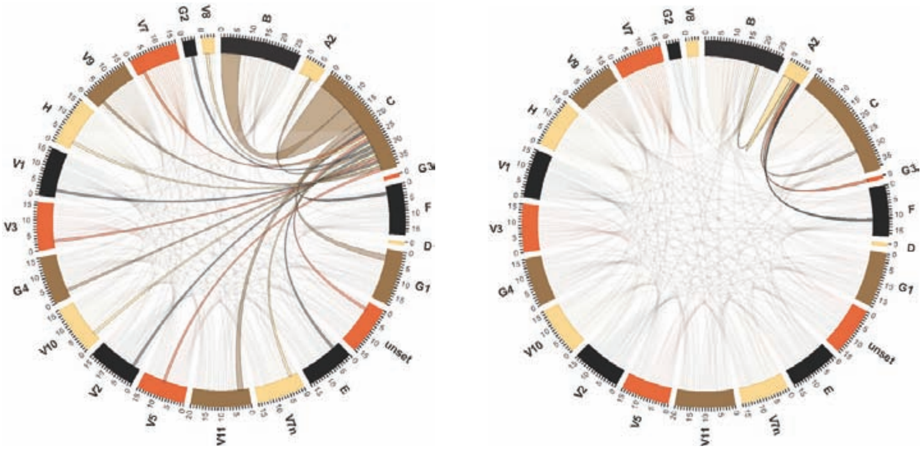


Fig. 21. Functions in organization lines. On the left side, the case of the organization line C is highlighted. On the right side, the case of the organization line A2 is highlighted.

3.2 Ping-Pong Behavior

The ping-pong behavior refers to the situation when the incident/problem handling is performed by bouncing tasks between STs, organizations or functions instead of actually handling the request. In the simplest situation, the ping-pong behavior consists in repeated interactions among two parties. In more advanced scenarios, the ping-pong might be performed among larger number of parties in cycles.

In order to detect cycles, a social network analysis approach was taken. A graph, referred to as the *cycle graph*, has been built to capture cycles in process instances. The relation among parties exists if in one process instance parties perform activities after one another. When relations create cycles, they are counted. Only direct sequences of activities are taken into account. It is possible that one cycle appears in one process instance many times.

Three perspectives were taken for our analysis of the ping-pong behavior: (1) product perspective (tackling the question which products generate cycles), (2) cycle perspective (tackling the question which cycles are shared among products), and (3) organizational structure perspective (tackling the question which organizational units contribute to cycles).

Product perspective. In the Incident Handling process, the products with the highest number of cycles are products 424 (107 cycles), 38 (49 cycles), 542 (55 cycles), 802 (cycles), and 776 (27 cycles). On the other hand, some products are supported with a very limited number of cycles, but with one dominant cycle, e.g., product 258 (9 cycles, cycle G179, D8, G179 appears 26 times, cf. Fig. 22), product 295 (4 cycles, cycle G358, S9, G358 appears 6 times), and product 350 (7 cycles, cycle T17, D1, T17 appears 15 times).

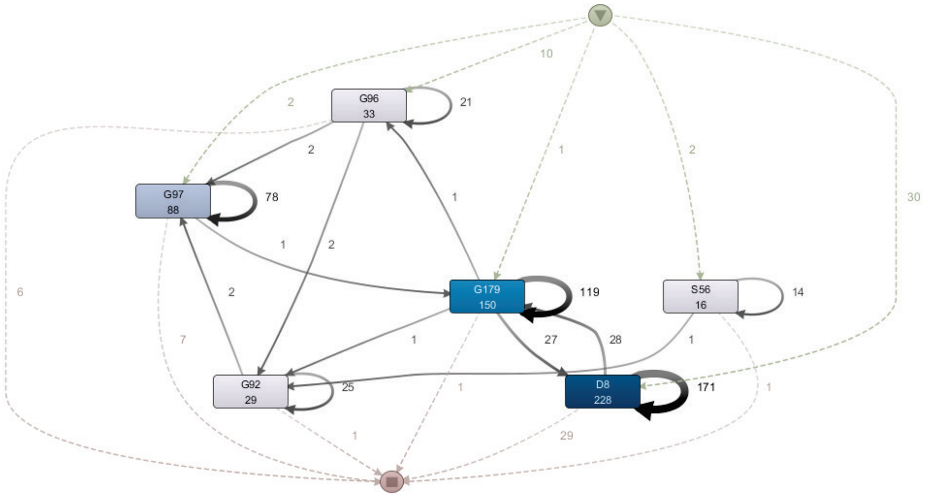


Fig. 22. Frequent cycle G179, D8, G179 for product 258.

Cycle perspective. The most popular cycle is D4, D5, D4 and it appears 57 times for products 236, 697, 791, 789, 235, 328, 542, 776, 158, 318, 92, 736, 312, 727, 799, 329, and 327. The process instances in which this cycle has occurred are presented in Fig. 23. The cycle between D4 and D5 is dominant. The numbers associated with activations and transitions indicate the number of process instances.

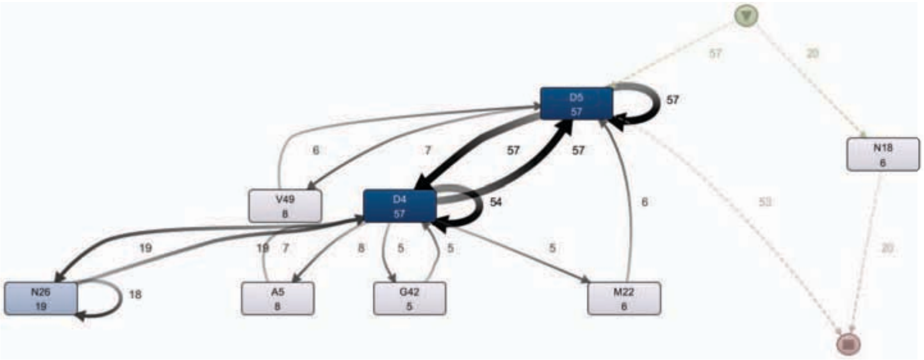


Fig. 23. Process instances containing the cycle D4, D5, D4.

Similarly, the cycle G179, D8, G179 has been recorded 40 times for products 258, 623, and 790 (Fig. 24). The third most common cycle is D4, N26, D4 (50 times, products 697, 791, 318, and 789).

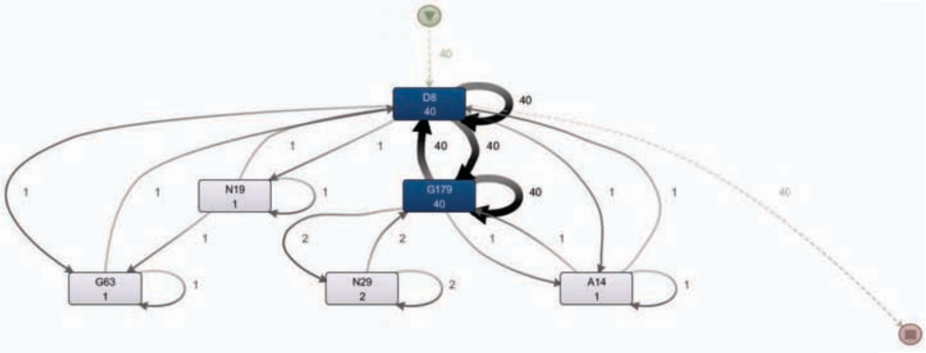


Fig. 24. Process instances containing the cycle G179, D8, G179.

In the Problem Handling process, the ping-pong behavior and cycles are not so common. Some cycles can be noticed in Closed Problems event log. The Open Problems event log is actually free of cycles. This confirms the observation that the Problem Handling process is definitely more structured.

There are two cycles in the Closed Problems event log. The first cycle G199, G21, G199 appears 25 times for products `97`, `98`, `96`. Process instances affected by this cycle includes only activities G199, G21 (cf. Fig. 25). The second cycle G88, G92, G88 appears 5 times for products `363`, `793`, `197`, `348`.

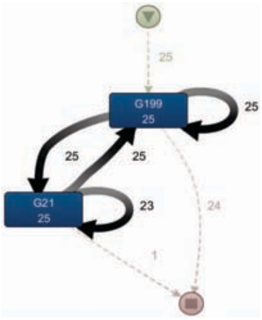


Fig. 25. Cycle G199, G21, G199 in Closed Problems event log.

Organizational structure perspective. In this perspective, the interaction among process owners and STs were analyzed.

Among process owners, the case of Siebel is particularly interesting. Seibel has the highest degree in the cycle graph, which means that among all the process owners participating into a cycle in a process instance, Siebel is the one connected to the highest number of unique process owners.

In the visualization of the cycle graph presented in Fig. 26, each node represents a process owner. The size of each node represents the degree of the process owner: the bigger the node, the higher the number of connected nodes. Therefore, a bigger node represents a process owner involved in cycles with a large number of process owners, while a small node represents a process owner involved in cycles with a small number of process owners.

The brightness of a node represents the betweenness of a process owner in the cycle of graph. Betweenness is defined as the ratio of the number of shortest paths through a given node between every two nodes and the overall number of shortest path between every two nodes in a graph. Betweenness represents the importance of a node with regard to information flows. The higher the betweenness of a node,

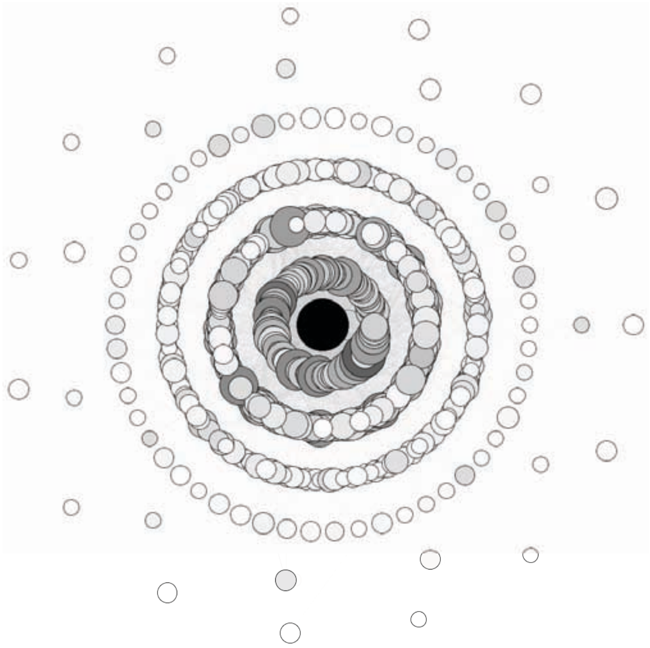


Fig. 26. Process owner cycle graph. The cycle graph centered on Siebel. The size of the nodes is proportional to their degree, the brightness is proportional to their betweenness centrality.

the more important it is in terms of transmission within the graph. In Fig. 26, the darker a node, the higher its betweenness. In the case of the cycle graph, process owners with a high betweenness are participating in a large number of cycles involving a large number of process owners, while process owners with a low betweenness are participating in a lower number of cycles with relatively few process owners.

Finally, nodes are laid out as follow in Fig. 26. The central node Siebel. Nodes located on the smaller circle are directly connected to Siebel. Nodes located on the second circle are directly connected to a node on the first circle. Therefore, nodes located on the second circle are two hops away from Siebel, that is they have participate in a cycle with someone who has been participating in a cycle with Siebel, or they have participate in a cycle with someone between them and Siebel.

The darkest node is Siebel. Siebel is the most central node in the network of cycles in term of betweenness. Most of the cycles linking two process owners are containing Siebel. Additionally, the degree of Siebel is the highest, and therefore, the node representing Siebel is the biggest.

Both the betweenness and the degrees of the nodes are fading down when going away from Siebel. The nodes are becoming brighter and smaller.

A similar analysis has been performed for STs and is presented in Fig. 27. The variety of colors and sizes among nodes corresponding to STs is wider that among process owners. The central node is ST G97. G97 belongs to organization C, the 1st support line and function division V3_2. It participates in many cycles with many various STs. Many cycles include G97.

Other STs are contributing to the existence of cycles (and therefore, to the ping-pong behavior): G96 (organization C, 1st line, function V3_2, in short C/1st/V3_2), G92 (C/1st/E_5), G271 (C/2nd/E_10), G179 (C/1st/V3_2), G230 (B/2nd/E_10), G40 (A2/1st/A2_2), D7 (A2/1st/A2_1), D2 (C/1st/A2_1), D5 (C/1st/A2_1), D8 (A2/1st/A2_1), D4 (A2/1st/A2_1), G51(G4/2nd/unknown).

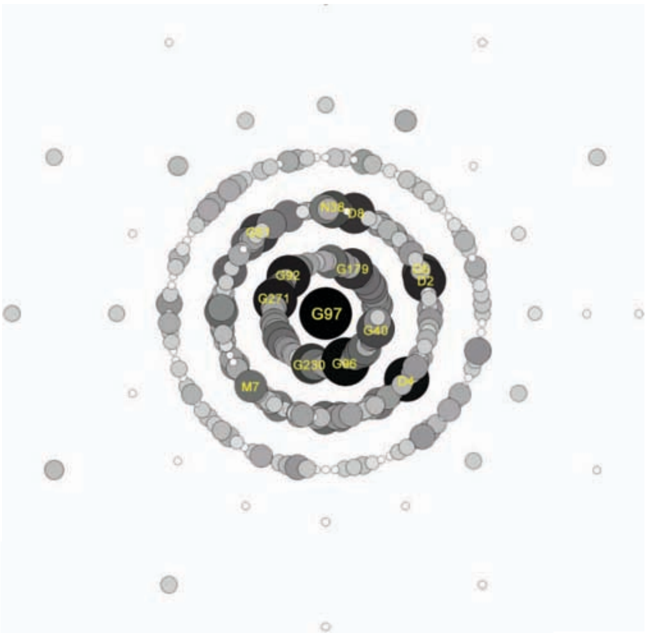


Fig. 27. ST cycle graph. The cycle graph centered on G97. The size of the nodes is proportional to their degree, the brightness is proportional to their betweenness centrality.

Organizations involved in most cycles are C and A2. Specifically, the cycle among those organization appeared 408 times in the incident event log. Cycle-prone organizations are B, V11, V7n, e.g., cycle C, B, C (29 times), C, V11, C (64 times), B, A2, B (44 times).

In the Problem Handling event log, cycles are generated by organization C, A2, G3 and B, e.g., cycles A2, G3, A2 (26 times), C, A2, C (10 times), C, B, C (9 times).

3.3 Wait User

The “Wait-user” sub-status does not appear in Problem Handling event logs at all. Therefore our analysis focuses on the Incident Handling process.

The “Wait-user” sub-status is always used with the “Accepted” status. The activity “Accepted/Wait-user” has been executed 4.217 times, which accounts for 6.43% of all the activities registered in the event log. The “Accepted/Wait-user” activity appeared in 2.495 (33%) process instances.

We have analyzed the “Accepted/Wait-user” activities with regard to products, STs, countries, function divisions, support lines, process owners, and organizations.

Products. Products 424, 660 are associated with the highest number of “Wait-user” sub-statuses. In Fig. 28, twenty products with the largest number of “Wait-user” sub-status are presented. The larger the circle, the higher the number of “Wait-user” sub-status. To evaluate potential overuse of the “Wait-user” sub-status, the ratio number of “Wait-user” sub-status /number of process instances has been calculated per product. In Fig. 28, the darker color of a circle, the higher the value of the ratio. A slight overuse of the “Wait-user” status can be noted for products 604 (“Wait-user” sub-status appeared in 84% of process instances), 262 (72%), 698 (69%), and 776 (67%). For those products, the ratio is definitely above the average of 45%. A relatively rare use of the “Wait-user” sub-status can be indicated for products 566 (21%), 13 (19%), and 267 (21%).

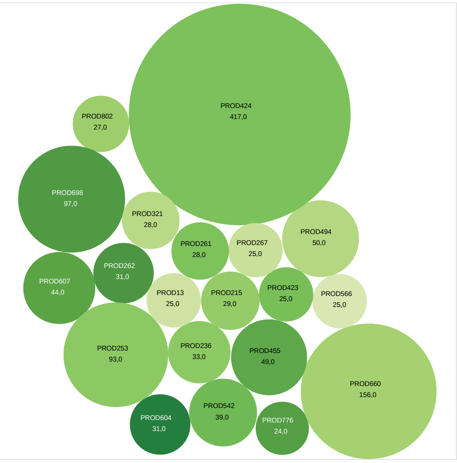


Fig. 28. “Wait-use” sub-status by products.

Support teams. STs that use the “Wait-user” sub-status the most frequently are presented in Fig. 29. In this square diagram, the size of a square represents the frequency of the “Wait-user”sub-status use. The color corresponds to the ratio between the number of “Wait-user” sub-status uses and the total number of the events.

The “Wait-user” sub-status has been executed 704 times by G97. The high number of “Wait-user” sub-status uses is justified by the large number of activities performed by G7. However, the “Accepted/Wait-user” activity accounts only for 9.43% of all the activities performed by G97.

The support teams G92 and S43 may be suspected of abusing the “Wait-user” sub-status: the “Accepted/Wait-user” activity accounts respectively for 16.55% and 15.2% of all the activities performed by those team. The higher than average use of the “Wait-user” sub-status is partially justified by the fact that all the activities performed by those STs are performed on the 1st support line, i.e., waiting for a user response is considered as a normal behavior on the 1st line. The “Wait-user” sub-status is rarely used by G96 (3%) and S42 (4.1%), which is surprising as G96 and S42 also perform all the activities in the 1st support line. In case of these STs, this leads to the conclusion that the two teams contribute to a better overall performance of the process.

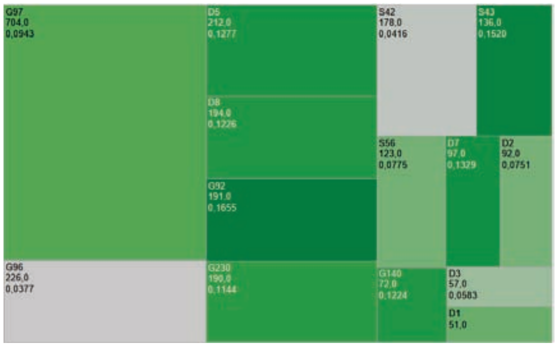


Fig. 29. “Wait-user” sub-status by support teams.

Countries. The ratio of the number of “Wait-user” status/the total number of activities per country is presented in Fig. 30. The sub-status is used most frequently in Sweden, Poland and India. In case of Sweden and Poland, it is justified by the large number of activities performed in those countries. However, India can be suspected for the overuse of the sub-status due to high ratio value. In India, “Accepted/Wait-user” activities accounted for more than 14% of all the activities performed in this country. This is far above the average.

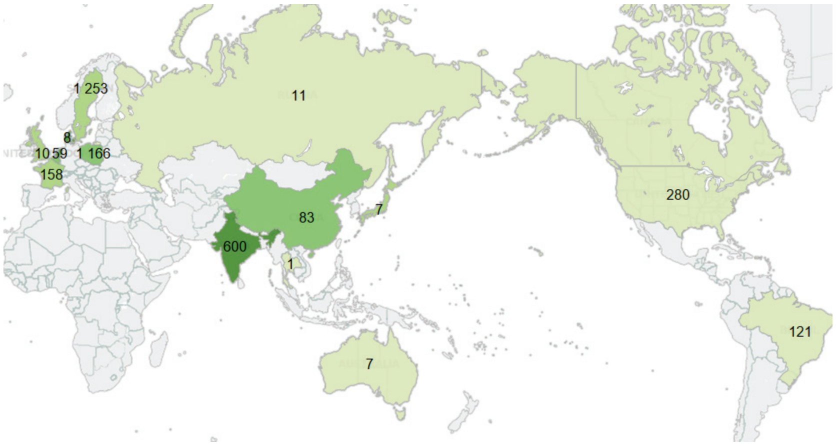


Fig. 30. “Wait-user” sub-status use by countries.

Function divisions. In Fig. 31, the use of the "Wait-user" sub-status is analyzed with regard to function divisions. None of the function divisions significantly distinguishes itselft from the others. The D_1 function division, with the highest use percentage (10.82%), is still close to the average. Moreover, D_1 performs the majority of it works in the 1st support line. Thus, the higher use of the "Wait-user" status is justified for D_1.

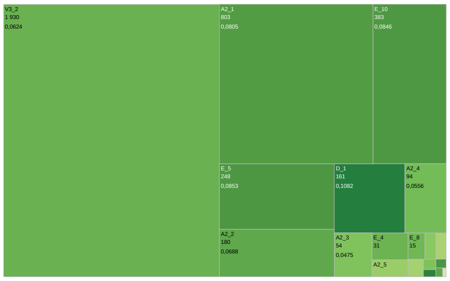


Fig. 31. "Wait-user" sub-status use by function divisions.

Support lines. When comparing support lines, the distribution of "Wait-user" sub-status uses among the three support lines follows the envisioned use of the "Wait-user" sub-status (cf. Fig. 32), i.e., the majority of sub-statuses appear in the 1st support line. The value of the ratio, 7.7%, is higher for the 3rd line but remains close to the average, i.e., 6.5%.

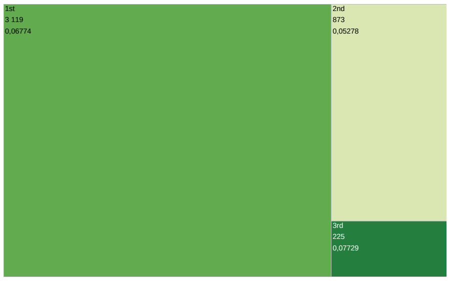


Fig. 32. "Wait-user" sub-status use by function divisions by support line.

Process owners. Process owners Muthu, Brecht, Emil, and Nina use the "Wait-user" status more often than the average process owner (Fig. 33), but as all of them specialize in performing actions on the 1st support line, it is rather a normal behavior.

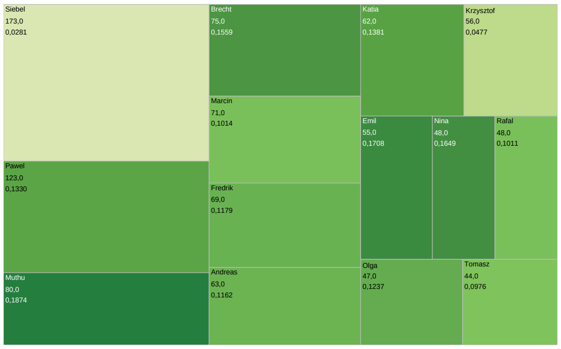


Fig. 33. "Wait-user" sub-status use by action owner.

Organizations. The use of the “Wait-user” status by various organizations is visualized in Fig. 34. Organizations are laid out according to the frequency of their use of the “Wait-user” status on the x-axis, and the percentage of their activities the “Accept/Wait-user” activity accounts for.

The difference between organizations A2, B, and C and the other organizations is due to the large volume of activities performed in general by A2, B, and C. However, two other organizations have to be distinguished: I and V1 (on the top of the figure). Those organizations perform a limited number of activities and almost one-fifth of these activities is the “Accept/Wait-user” activity.

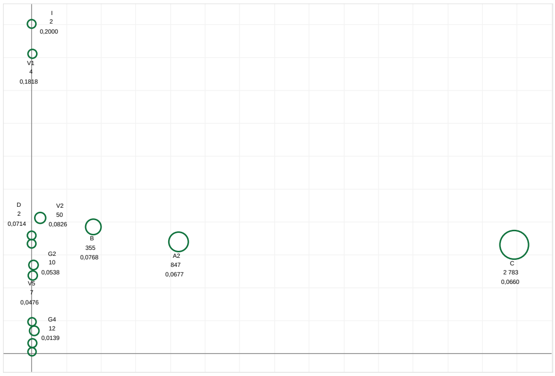


Fig. 34. “Wait-user” status use by organizations.

3.4 Process Conformity per Organization

The conformity of organizations with the assumed process model provided in [1] was checked on a basis of the Incident Handling process dataset. Activities are here described by status and support line number, e.g. “Accepted-1st”.

Process Conformity for Organization C. Organization C performs 29.184 activities in 4.417 process instances. A model of the behavior recorded for 3.879 (87%) process instances involving C is presented in Fig. 35. Exactly 240 variants of process execution have been discovered for this subset of process instances. In Fig. 35, the numbers correspond to the number of process instances a given activity or transition has been executed in. Most process instances have been executed in the 1st line and included two activity types: “Accepted-1st”, “Completed-1st”.

Although most process instances are conform with the assumed process model, some transitions visible in the figure should not appear according to the model, e.g., transitions from activities performed in the 1st line to the 3rd line, transitions from the 3rd line to the 2nd.

The figure does not include 538 (13%) process instances that are absolutely unique and do not follow the model. Those are process instances involving multiple activities and are characterized by a long duration of execution.

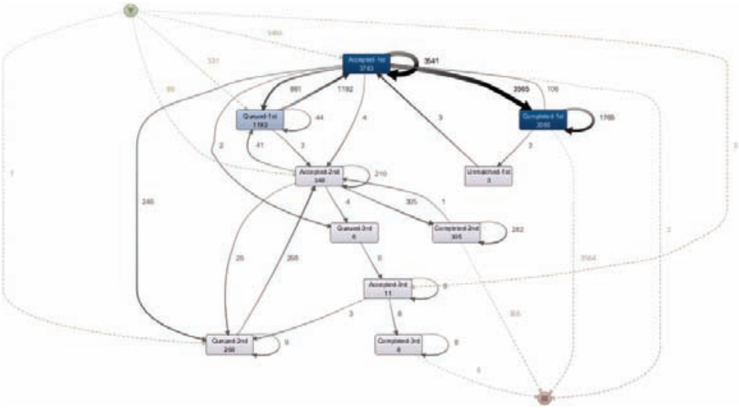


Fig. 35. Mainstream behavior in organization C.

Process Conformity for Organization A2. A similar figure for organization A2 is presented in Fig. 36. The organization A2 has executed 4.306 activities within 553 process instances. A model of the behavior recorded for 440 (79%) process instances involving A2 is presented in Fig. 36. 74 variants of process execution have been discovered for this subset of process instances.

Most process instances are executed by organization A2 in the 2nd line. Most frequently process instances include activities "Accepted-2nd" and "Completed-2nd". This is clearly visible in Fig. 36.

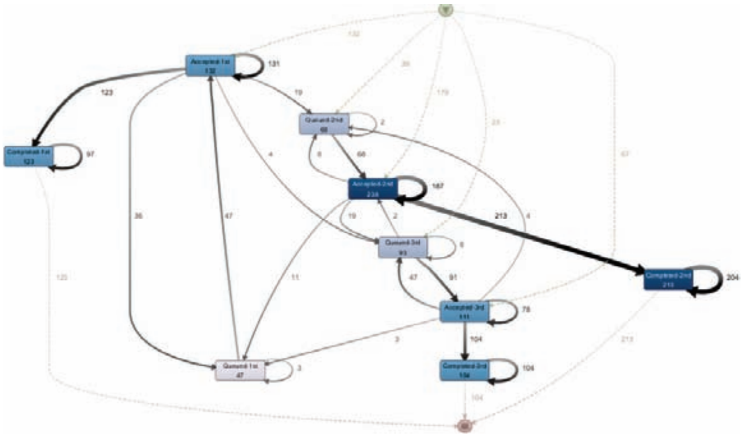


Fig. 36. Mainstream behavior in organization A2.

Comparing Process Conformity in Organizations C and A2. More that 80% of activities are performed by organization C in the 1st line. In case of organization A2 it is less that 40%. Missing adherence to push to front strategy is a serious problem for organization A2. The difference in this aspect between A2 and C is visible in Fig. 37.

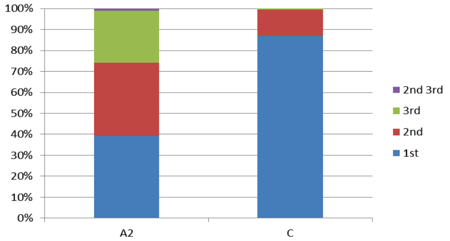


Fig. 37. Activities performed by organizations C and A2 in various support lines.

Fig. 36 does not include 113 process instances that are absolute unique and explain the lower efficiency of organization A2 operation. In case of organization A2 those exceptional process instances account for 21% of all the process instances. This is about 8% more than for organization C.

Concluding, *organization C is more in line with model of operations than organization A2*. Organization A2 forwards too much of its operation to the 2nd and 3rd line, and the execution of the process instances is much less standardized leading to the high percentage of unique executions.

4 Tools

4.1 Data munging

The basic tool for data munging used for our analyses has been *Microsoft Excel 2010*. Its capacities to import CSV file have allow the original datasets to be imported to a tabular format. Next, filtering and sorting capabilities of MS Excel have eased the identification of missing data and have allowed for raw estimations of the count and distribution of the data, such as the number of organization lines.

4.2 Data processing

An important tool for our analyses is the *Python language* and the *Enthought Canopy environment*. Python, in its version 2.7.4, has been used to process the cleaned data to obtain a better understanding of the concept of lines and the relation between lines and process owners/organization lines/functions. The data presented in Tabs. 3, 4, 5, and 6 have been computed in Python.

The following Python modules have been useful for our analyses:

- the *csv* module has been used to import and export data from and to CSV files,
- the *networkx* module has been used to create and analyze graphs, especially the cycle graph used to study ping-pong.

The Enthought Canopy has been chosen as an environment supporting of-the-box the networkx module and encompassing the IPython environment providing support for Python notebooks.

4.3 Process Mining Tools

For visualization of control flow in both Incident and Problem Handling processes, *Disco* (`http://fluxicon.com/disco/`) has been used in version 1.3.6. Disco is a commercial software developed by the Fluxicon company. The academic license allows analysis of event logs up to 1 million events.

The process map generator (cf. Fig. 1), build-in filtering mechanism, import of CSV files and export to MXML and XES formats compatible with ProM 5 and ProM 6 applications has been used in our analyses. Disco's built-in filtering algorithm allows including or excluding process instances and activities based on the appearance of one or more properties. Although Disco does not provide support for process conformance checking, the filtering algorithms can be used to fill this gap. This is especially useful when testing various hypotheses.

The *ProM Framework* (`http://www.promtools.org/prom6/`) is an open source application for process mining. ProM perfectly complements the functionality of Disco. ProM 6 has been used to perform social network analysis (*hand-over of work* relations were examined). Dotted chart diagram (e.g., Fig. 2b) have been generated with ProM 5.

4.4 Data visualization

The visualization of chord diagrams is based on the *D3.js* javascript library, available at `http://d3js.org/`, and the examples of chord diagrams published by Mike Bostock (`http://bl.ocks.org/mbostock/4062006` and `http://bl.ocks.org/mbostock/1046712`).

The visualization of the cycle graph has been performed in *Gephi* 0.8.2-beta (`http://www.gephi.org/`). The processing of betweenness and degree centralities has been performed with Gephi, as well as the laying out process and the final rendition of the graph visualization.

Square (e.g., Fig. 33), circle (e.g., Figs. 28 and 34) and map diagrams (e.g., Fig. 30) were generated using *Tableau*. Tableau is a commercial software (15 day trial is available) used for visualization of data (`http://www.tableausoftware.com/`). The software significantly improves the process of finding patterns in analyzed data by providing the possibility of fast and convenient switching among various data visualizations.

5 Conclusions

The Third International Business Process Intelligence Challenge is an interesting opportunity to check the potential use of various available tools to dig into real-life data. The size of the datasets was relatively small. As a consequence, all our analyses have been performed on personal computers and do not require heavy data processing infrastructure. Analyses of the full dataset of the VINST system event logs require probably different tools than those used to perform the analyses described in this paper.

The datasets provided by Volvo IT Belgium is especially interesting because of the complex organizational structures (STs, organization lines, support lines, function lines) that provide support for the Incident and Problem handling processes. We have chosen to address this aspect of the datasets with the toolkit established by the social network analysis (SNA) perspective community. It seems to us that the existing tools, such as the Python *networkx* module and *Gephi*, are able to provide real insights to complex social settings. However, expertise in SNA is a strong requirement for these tools that requires an important human involvement to extract interesting knowledge from the datasets.

Process mining and SNA have shown to be complementary: process mining tools have provided insights about the dynamic character of the Incident and Process Handling processes, while SNA tools have provided insights about the social/organizational structures in which these processes are embedded. With the intensification of research efforts on temporal networks, SNA may with no doubt benefit from the results of the works of the process mining community.

The image of the implementation of Incident and Problem Handling processes by Volvo based on our analyses is rather positive: issues are usually localized to given STs, products, or organization lines. The push to front strategy is rather well respected, with the expectation of the 4th and 5th lines that we have identified during our analyses. Ping-pong behavior is rather marginal. The mis-use of the "Wait-User" sub-status is restricted mainly to few organizations. Finally, although organization A2 conforms less with the proposed process model than the organization C, most process instances are close to the proposed model.

Among future works, it would be interesting to check if our results are confirmed by larger VINST event logs.

Acknowledgments. This work has been partially supported by the Polish National Science Center. Grant no. DEC-2011/01/N/ST6/04205.

References

1. Steeman, W.: VINST information needed to understand the dataset. Available online at `http://www.win.tue.nl/bpi2013/lib/exe/fetch.php?media=vinst_data_set.pdf` (2012) [last access: July 11th, 2013].
2. ITIL Foundation: Official ITIL website. Available online at `http://www.itil-officialsite.com` (2013) [last access: July 11th, 2013].
3. Van der Aalst, W.: Process Mining: Discovery, Conformance and Enhancement of Business Processes. Springer, Heidelberg (2011)
4. Jackson, M.O.: Social and Economic Networks. Princeton University Press (2008)
5. Steeman, W.: Incident management event log. Available online. doi: `10.4121/uuid:500573e6-accc-4b0c-9576-aa5468b10cee` (2013) [last access: July 11th, 2013].
6. Steeman, W.: Problem management log (open problems). Available online. doi: `10.4121/uuid:3537c19d-6c64-4b1d-815d-915ab0e479da` (2013) [last access: July 11th, 2013].
7. Steeman, W.: Problem management log (closed problems). Available online. doi: `10.4121/uuid:c2c3b154-ab26-4b31-a0e8-8f2350ddac11` (2013) [last access: July 11th, 2013].