Analysis of a collaborative workflow process with distributed actors

Hajo A. Reijers · Minseok Song · Byungduk Jeong

Abstract Current workflow management technology offers rich support for process-oriented coordination of distributed teamwork. In this paper, we evaluate the performance of an industrial workflow process where similar tasks can be performed by various actors at many different locations. We analyzed a large workflow process log with state-of-the-art mining tools associated with the ProM framework. Our analysis leads to the conclusion that there is a positive effect on process performance when workflow actors are geographically close to each other. Our case study shows that the use of workflow technology in itself is not sufficient to level geographical barriers between team members and that additional measures are required for a desirable performance.

Keywords Workflow management ·
Performance evaluation · Process mining ·
Case study

H. A. Reijers (✉) · M. Song
Department of Technology Management,
Eindhoven University of Technology,
P.O. Box 513, 5600 MB,
Eindhoven, The Netherlands
e-mail: h.a.reijers@tue.nl

M. Song
e-mail: m.s.song@tue.nl

B. Jeong
Department of Industrial Engineering,
Korea Advanced Institute of Science and Technology,
Daejeon, South Korea
e-mail: tauruseo@kaist.ac.kr

1 Introduction

Workflow technology is considered as one of the prime enablers for team members to work on complex tasks while residing at different work places (Leymann and Roller 2000; Rinderle et al. 2004). Yet, surprisingly little is known about the *extent* to which workflow technology helps organizations to execute their business processes more efficiently and effectively, let alone when the involved actors collaborate in *a distributed setting*.

To fill the gap in knowledge on the organizational effectiveness of workflow management systems (WfMSs) a joint research project was initiated in 2001 by Eindhoven University of Technology and Deloitte Consultancy. The purpose of the project is to closely monitor a large number of organizations that implement and use WfMSs to support their operational business processes. The details on the set-up of this project and an overview of its preliminary results can be found in Reijers and van der Aalst (2005). The project, in which 10 organizations are involved and over 20 business processes, is still ongoing and is expected to be completed in 2008. This paper builds on insights that were developed in the context of one of the participating organizations in this project.

The particular organization of interest is a local Dutch municipality, which started the implementation of the WfMS Staffware for their invoice handling process in 2003. In the second half of 2004, they went "live". Through our collaboration with the municipality we acquired access to the WfMS's process log which contained the registered events for 2005's production, covering over 12,000 completed cases in total. The process in question involves the handling of invoices

that the municipality receives from its contractors. As will be explained in more detail in this paper, a quite complicated approval procedure is in place to decide whether invoices are to be paid. The most interesting thing of the invoice handling process is that the collaborating parties are distributed over 10 different locations, all across the municipality's geographical region. Moreover, there are tasks which can take place in any of these locations. After all, an invoice can pertain to almost anything (e.g., pencils, pc's, or furniture) and must be checked by a responsible civil servant who can work at the city's fire brigade, swimming pool, theater, or any other part. The unique characteristic of this setting is that for each new instantiation of the process the involved actors may be at *different* geographical locations while the steps they are perform from a business perspective are the *same*.

The described setting provided a rare opportunity to investigate the effect of a geographical dispersion of workflow actors on the performance of the overall process, by using the event logs we had access to. Specifically, this paper's contribution is that it gives an empirical insight into the question whether it matters for *process performance* when collaborating actors in a business process, supported by *workflow technology*, are *geographically distributed*. As will be explained further on, there is a tendency to believe that workflow technology will make it less relevant where actors actually reside physically, but this paper raises concerns about the validity of this belief. An additional contribution is that this paper provides an empirical yet quantitative evaluation of a workflow implementation. This type of evaluation is rather rare in the academic discourse, but seems in much demand to test the beliefs about the effectiveness of workflow technology.

The remainder of this paper is organized as follows. We provide an overview of related work in Section 2. In Section 3 we will describe our research design, in particular the hypotheses we set out to investigate. Then, we will describe the case study in more detail in Section 4, along with our analysis and findings. The paper provides a discussion of the results in Section 5, including the limitations of our study. Finally, the concluding remarks are given in Section 6.

2 Related work
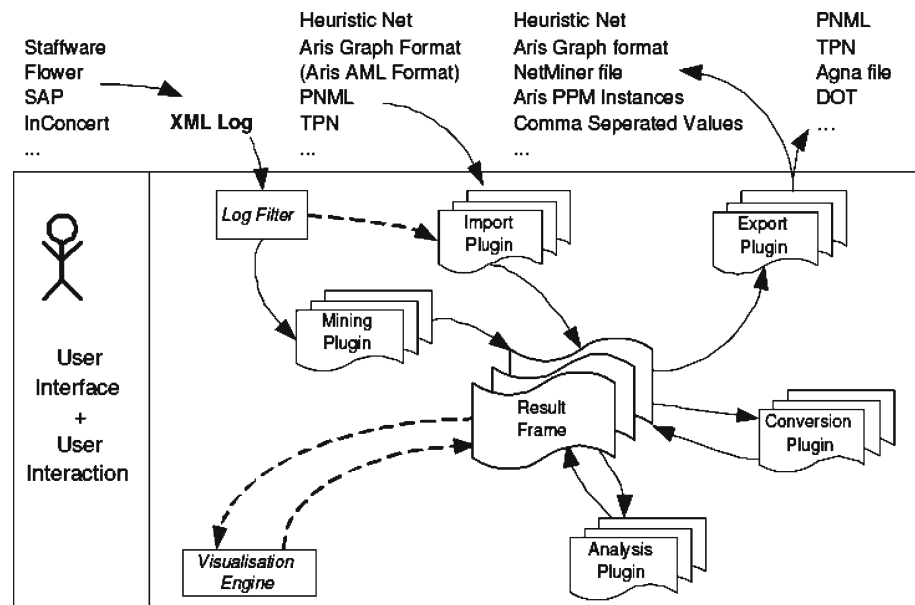
2.1 Workflow and geography

In principle, business processes can be executed faster and more efficiently by using a WfMS for the logistic management of a business process (Lawrence 1997). WfMS vendors and market analysts claim that these advantages materialize in practice, see e.g. (Palmer 2007; Staffware 2000). In academic papers, various single case studies of workflow implementations are described and a small number of studies that involve multiple implementations (Herrmann and Hoffmann 2005; Kueng 2000; Oba et al. 2000). Most of the studies that explicitly consider performance established a positive effect of workflow technology, in particular in Oba et al. (2000). However, none of these studies examined whether the geographical distribution of actors played any part in such performance improvement. (Note that the *architectural* issues that relate to distributed workflow processing have been widely studied, e.g. in Grasso et al. (1997), Vonk and Grefen (2003), Grefen et al. (2003), Fakas and Karakostas (2004), Blake and Huhns (2008)).

In his seminal work, Allen (1977) reported that geographical distances between actors may indeed be important in some contexts. In the late 1970s, Allen undertook a project to determine how the distance between engineers' offices coincided with the level of regular technical communication between them. The results of that research, now known as the *Allen Curve*, revealed that when there is more distance between people they will communicate less frequently.

However, there is a widely felt belief that due to the massive utilization of information and communication technologies (ICTs) the precise physical location of individual participants will become irrelevant to their interactions (Boutellier et al. 1998; Carmel and Agarwal 2001). ICTs are a key enabler for the emergence and sustained popularity of so-called *virtual teams*, i.e. groups of geographically and organizationally dispersed coworkers that are assembled using a combination of telecommunications and information technologies to accomplish an organizational task (Townsend et al. 2000). WfMSs too enable the fast communication and collaboration between geographically dispersed users and can therefore be expected to contribute to improved interaction between them (Becker and Vossen 1996; Sengupta and Zhao 1998; Steinfield et al. 1999). In particular, in van der Aalst and van Hee (2002) it is stated that "The introduction of a WfMS lowers the physical barriers between the various sections of an organisation". It continues to state that a WfMS can, for example, be used to more evenly distribute work among geographically scattered resources. Therefore, the image that emerges from contemporary literature on ICTs in general and workflow technology in particular is that it has become less relevant where people reside physically for the performance of a

Fig. 1 Overview of the ProM framework (van der Aalst et al. 2007b)

collaborative process. Concrete evidence for this belief is yet missing.

2.2 Process mining

In this paper, process mining techniques are applied to analyze business process execution results. Process mining allows the discovery of knowledge based on a process log (van der Aalst et al. 2004). Process logs, which are provided by most process aware information systems, records the execution of tasks in some business processes. Process mining deals with several perspectives such as the process perspective, organizational perspective, performance perspective, etc.

To support process mining, various researchers have developed several tools (van der Aalst et al. 2004, 2005, 2007b; van Dongen and van der Aalst 2004; Herbst and Karagiannis 2004; IDS Scheer 2002). From these, we will use the ProM framework to analyze process logs. The ProM framework has been developed to support various process mining algorithms. It was designed to easily add new algorithms and techniques into the ProM framework by means of plug-ins (van der Aalst et al. 2007b). A plug-in is basically the implementation of an algorithm that is of use in some part of the process mining area.

Figure 1 shows an overview of the framework. It reads log files in the XML format through the *Log filter* component. This component can handle large data sets and sort the events within a case according to their time stamps. Through the *Import plug-ins* a wide variety of models can be loaded ranging from Petri nets to logical formulae. The *Mining plug-ins* perform

the actual process mining. The *Analysis plug-ins* take a mining result and perform a further analysis. The ProM framework provides several analysis techniques such as Petri-net analysis, social network analysis, performance analysis, etc. The *Conversion plug-ins* can convert a mining result into several other formats, e.g., transforming an EPC into a Petri net. The *Export plug-ins* can export the mined results, filtered logs, etc.

3 Research design

This section explains our research questions and describes the research method with which they are addressed. We established a research procedure as shown in Fig. 2. The remainder of this section explains each step in the context of our research.

Research questions This study aims to determine how the performance of a business process is affected by the use of a WfMS in a geographically distributed setting. To explain our research questions, the two process performance indicators selected to investigate should be introduced. They are defined as follows:

– *Processing time*: the time between the start of a task and its completion,
– *Transfer time*: the time between the completion of a task and the start of a subsequently executed task

Since a WfMS takes care of delivering the right work to the right person at the right time, the use of such a system can be expected to result in a reduction of process throughput times. No longer is the individual
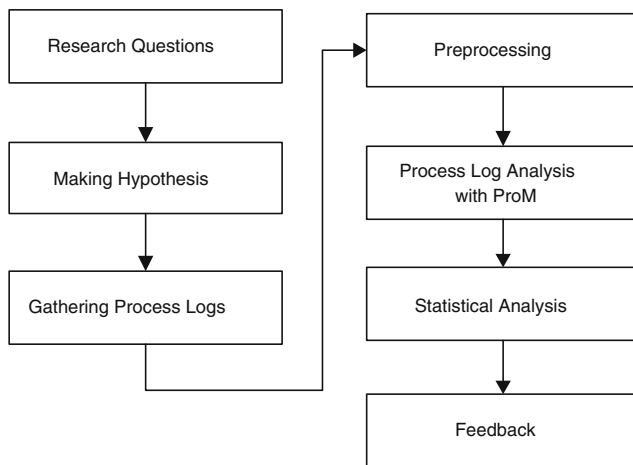
Fig. 2 The research procedure

worker burdened with the task to collect all relevant information and to decide how a work package must be routed further through an organization. When a WfMS takes care of assigning work to actors, it is therefore perhaps less relevant where these actors are located geographically. When companies introduce WfMSs, they normally perform business process re-engineering projects. During such projects, as-is analyses are carried out and geographical influences in the execution of business processes are removed by standardizing its tasks. After that, multi-functional teams that involve business professionals, information analysts, and system integrators, design new business processes and implement them with WfMSs. In a WfMS, when a task is completed, the following task is immediately assigned to a proper actor, i.e. it is added to the worklist of the actor regardless of his/her geographical location. Next, it is handled by the actor. Thus, it appears so that the introduction of workflow technology makes geographical influences irrelevant. These ideas lead us to the following two research questions.

– *Research question 1*: How is *processing time* affected by workflow technology in terms of the geographical location of actors?
– *Research question 2*: How is the *transfer time* affected by workflow technology in terms of the geographical location of actors?

Making hypothesis From the *research questions*, the first step is *making hypothesis*. Our research questions lead to the formulation of two hypotheses:

– *Hypothesis 1*: The *processing time* of equivalent tasks is equally distributed, despite the geographical locations in which the tasks are performed.

– *Hypothesis 2*: The *transfer time* between tasks within the same geographical location is equally distributed as the transfer time between tasks across geographical locations.

Hypothesis 1 deals with the first research question. For the hypothesis, we considered tasks that can be performed in several geographical locations. We calculated processing times of tasks within each location and compared them. *Hypothesis 2* addresses the second question. In this case, we considered the pairs of tasks that can be successively executed in the same geographical location or in different geographical locations.

Gathering process logs The next step is *gathering process logs* used to examine our hypotheses. We gathered process logs from the involved organization, which uses the software package Staffware as its WfMS. From its database, we extracted event logs covering six months of operation, which seems like an adequate time period to test our hypotheses. The more information on the process logs are explained in Section 4.2.

Preprocessing The next step is *preprocessing* the logs (Section 4.2). Since the process logs gathered were stored in a proprietary format, we had to preprocess the process logs. They were converted into a standard MXML format (van der Aalst et al. 2007b).

Process log analysis with ProM After the conversion, we *analyzed the logs with the ProM framework and its associated tools*. This step consists of two phases: the *initial analysis* (Section 4.2) and the *further analysis* (Section 4.3). At the initial analysis, we calculated the overall performance (i.e. execution time) of each task and derived a process model from the entire process logs. Then we selected target tasks for the further analysis. For this further analysis, we removed irrelevant tasks and calculated relevant processing and transfer times.

Statistical analysis The next step is *statistical analysis* (Section 4.4). We performed various statistical tests to examine our hypotheses. Since the ProM framework does not support statistical analysis, we generated the data for statistical analysis from the ProM and used Statgraphics Centurion XV for the tests.

Feedback To evaluate our research results, the analysis results from the ProM and the statistical test results were reported to the organization that provided the logs and discussed. Finally, we *gathered their feedback* (Section 5).

4 Case study

4.1 Context

Our case study involves in the Urban Management Service of a local municipality of 90,000 citizens, situated in the northern part of the Netherlands. The municipality is one of the organizations that is involved in our longitudinal study into the effectiveness of workflow management technology (Reijers and van der Aalst 2005). In 2000, the board of the municipality decided to implement a WfMS throughout the organization, which encompasses some 300 people. Mainly because of restricted budgets and some technical setbacks, it lasted until 2004 before the first two business processes were supported with this technology. One of these two processes involves the handling of invoices, which is the focus of our analysis.

Every year, the municipality deals with about 20,000 invoices that pertain to everything that the municipality purchases. The overall process consists of 26 different tasks and may involve almost every employee of the Urban Management Service. After all, an important check is whether the invoice is 'legitimate' in the sense that it corresponds with an authorized purchase by some employee, to be checked by that employee himself/herself. Also, various financial clerks play a role in this process. It is important to explain here that the municipality both has a *central* financial department and various *local* financial departments attached to its sectors, i.e. its divisions. The *sectors* are distributed over all the geographical locations of the municipality (e.g. the mayor's office, the city's swimming pool, the fire brigade, etc.). We will now give a simplified description of the general procedure, including labels for some of the most important steps between brackets.

When an invoice is received, it is scanned by a member of the municipality's central financial department and subsequently registered in the financial system. If the invoice's creditor is not yet known to the municipality, a new record is created. In most cases, the scan is legible and can be routed further to one of the various local financial departments of the organization – in case it is not, it is re-scanned. The next step is then that a clerk from one of the local financial departments of the municipality must evaluate whether the invoice is indeed intended for this sector (ROUTEFEZ). Sometimes, it is difficult to determine for which sector an invoice is intended, particularly if reference information is missing from the invoice. A case may be routed from the local financial department of one sector to another until it arrives at the right place. When this is so, a so-called budget keeper – the person responsible for the

budget within a sector that is used for the purchase – must subsequently check whether the invoice can be approved and adds a code to the invoice that expresses the outcome of this check (CODFCTBF). If the budget keeper did not make the purchase himself/herself, then he or she can decide to route the invoice to the colleague who did. Such a colleague usually works within the same sector, but this is not always the case. The latter person must then check the invoice and add a decision code too (CONTRUIF). Then, the invoice is routed back to the local financial department of the sector where the budget keeper belongs to, where a clerk then checks the given decision code(s) (CONTRCOD). When the invoice has satisfactorily been dealt with, the invoice is routed back by this clerk to the central financial department. If the invoice amount exceeds certain standards, further approval may be required from a senior clerk (BEOORDSR). The senior clerk may then even decide to have the invoice checked additionally by the financial department head and/or the head of the involved sector. Whether such additional checks have been carried out or not, a clerk of the central financial department must eventually check all the assigned codes (FBCONCOD). The invoice is then either paid or not by the central financial office.

What is important to stress here is that by Dutch law, governmental bodies need to pay their invoices within 30 days or risk financial penalties. This explains in part why the municipality's board was interested to automate this process in the first place. The other reason is the wide distribution of the various actors in this process, which makes it difficult to control without a WfMS.

4.2 Process log and initial analysis

This section describes the process log and some initial process mining results. To help us better understand the process log, we examined the overall statistics of the process log and investigated some conventional process mining results, such as a dotted chart analysis and a mining of the control flow structure. These will be addressed in more detail.

In the case study under consideration, a process log is automatically generated by the WfMS executing the invoice handling process. The process log gathered from the organization was translated into a MXML format, so that the ProM framework could import it. A process log consists of several *instances* or *cases*, each of which may comprise several audit trail entries. An *audit trail entry* corresponds to an atomic *event* such as schedule, start, or completion of a task. Each audit trail entry records the task name, event type,

Fig. 3 Fragment of the example log in MXML format

```
<?xml version="1.0" encoding="UTF-8"?>
<WorkflowLog xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xsi:noNamespaceSchemaLocation="http://is.tm.tue.nl/research/processmining/WorkflowLog.xsd">
    <Source program="Staffware"/>
      <Process id="Facturen" description="none">
            <ProcessInstance id="4-21334" description="none">
            <AuditTrailEntry>
            <WorkflowModelElement>PREREG</WorkflowModelElement>
                <EventType>schedule</EventType>
                <Originator>org1</Originator>
                <Timestamp>2005-07-13T13:55:00+01:00</Timestamp>
            </AuditTrailEntry>
            <AuditTrailEntry>
            <WorkflowModelElement>PREREG</WorkflowModelElement>
                <EventType>start</EventType>
                <Originator>system1</Originator>
                <Timestamp>2005-07-13T13:58:00+01:00</Timestamp>
            </AuditTrailEntry>
            <AuditTrailEntry>
            <WorkflowModelElement>PREREG</WorkflowModelElement>
                <EventType>complete</EventType>
                <Originator>org1</Originator>
                <Timestamp>2005-07-13T14:01:00+01:00</Timestamp>
            </AuditTrailEntry>
            ...
```

actor and time stamp. Figure 3 shows the example of a translated process log in MXML format. It is interesting to note here that we had to develop additional code and install it at the workflow server to record the exact times that tasks were initiated by the involved people. Oddly, the Staffware system only records by default the time that a work item is made available to a group of workers, i.e. scheduled, and the time that a work item is completed by a particular worker. Clearly, from the two latter types of information alone, exact processing times cannot be determined.

The process log we analyzed covered slightly more than 12,000 instances (completely handled invoices), as processed by the municipality in the first half of 2005.

To have an insight in the overall events, we performed the dotted chart analysis that shows the distribution of events over time. Figure 4 depicts a screenshot of the *dotted chart analysis* plug-in, where each row corresponds to one of the process instances displayed on a time scale. In the diagram, we use the notion of *relative time*, which shows the duration from the beginning of an instance to a certain event. Thus, it indicates the case duration of each instance. In the diagram, we can easily recognize that process instances have more events at their beginning stage. Many of the cases finished within two or 3 months. More precisely, the average case duration is 16 days. Where the shortest case took only 1.3 hours, the longest case took more than 5 months.

Fig. 4 Dotted chart analysis: the distribution of the events over time
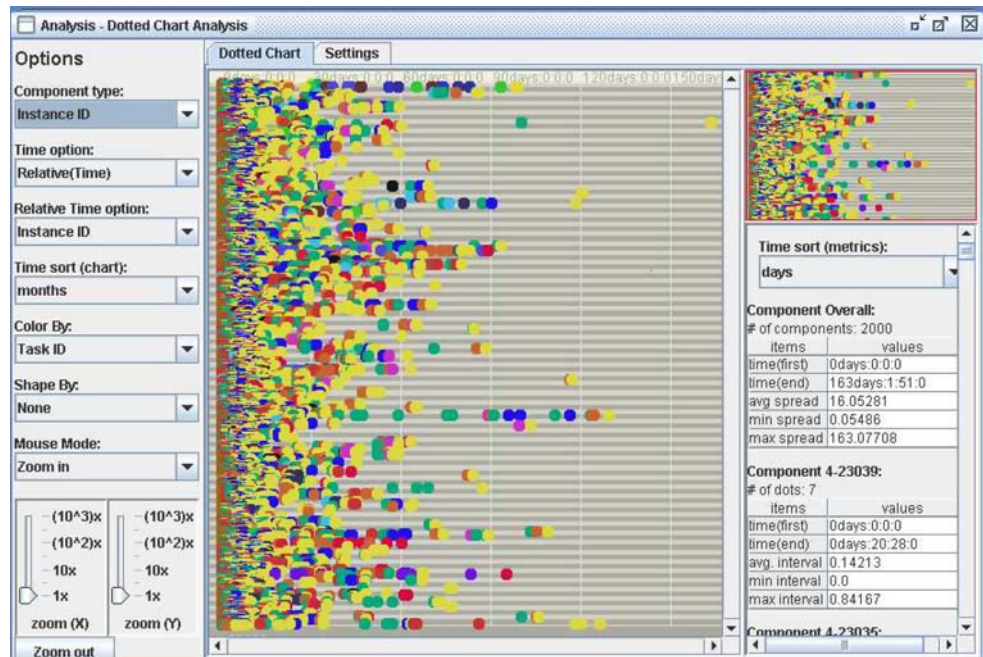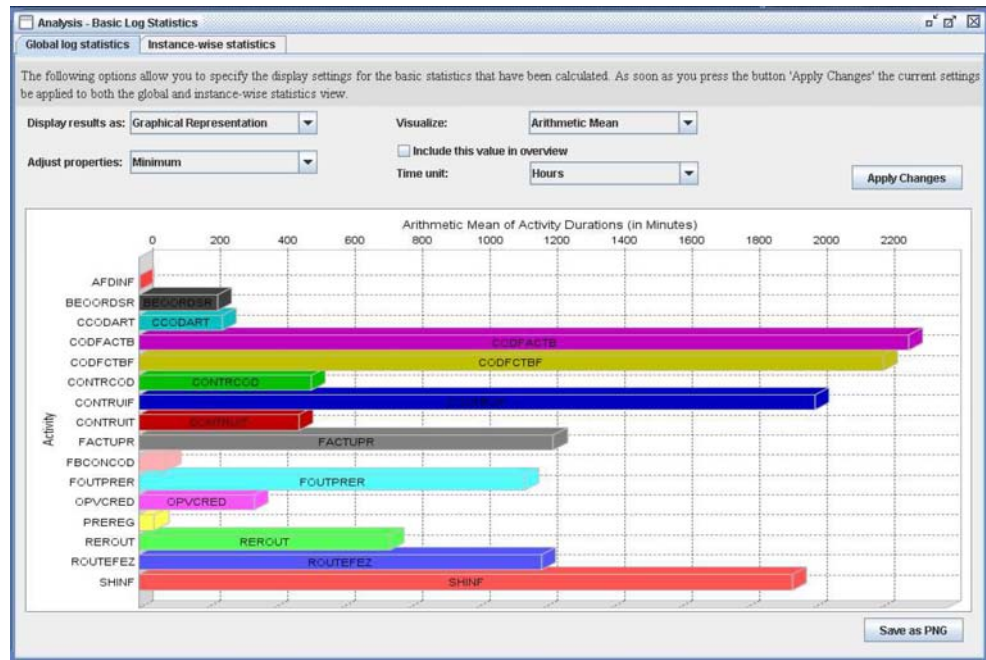
Fig. 5 Basic log statistics

To inspect each task duration of the process log, we used the *basic log statistics* plug-in. The plug-in evaluates the 'start' and 'completion' times for each task in the process. This way, steps in the process that consume much time can be detected as further analysis candidates. Figure 5 depicts a screenshot of the graphical view on the mean time that is spent within each task. The average durations are diverse from few hours (e.g. BEOORDSR, CCODART, etc.) to more than a day (e.g. CONTRCOD, CODFCTBF, etc.).

To see the behavior (i.e. control flow) of the process log, we performed *control flow mining*, which automatically derives process models from process logs. The generated process model reflects the actual process as observed through real process executions. Until now, there are several process mining algorithms such as the α-mining algorithm, heuristic mining algorithm, region mining algorithm, etc. (van der Aalst et al. 2004; Weijters and van der Aalst 2003; van Dongen et al. 2007). In this paper, we use the heuristic mining algorithm, since it can deal with noise and exceptions, and enables users to focus on the main process flow instead of on every detail of the behavior appearing in the process log (Weijters and van der Aalst 2003).

The heuristic mining algorithm considers dependency relations between tasks and derives a process model using these relations. It uses a frequency based metric which indicates how frequently there are dependency relations between two tasks A and B (notation $A \Rightarrow B$).

Let T be a set of tasks, $\sigma \in T*$ be a workflow trace and $L \in P(T*)$ be a process log over T, and $a, b \in T$:

- $a >_L b$, if and only if there is a trace $\sigma = t_1 t_2 t_3 ... t_{n-1}$ and $i \in 1, ..., n - 2$ such that $\sigma \in L$ and $t_i = a$ and $t_{i+1} = b$,
- $|a >_L b|$ is the number of times $a >_L b$ occurs in L (i.e., the number of times event a is directly followed by event b),
- $a \Rightarrow_W b = (|a >_W b| - |b >_W a|) / (|a >_W b| - |b >_W a| + 1)$.

For example, if task A is directly followed by task B five times in a log, but the reverse case occurs only once, the value of $A \Rightarrow_W B$ equals $0.571 (= (5-1)/(5+1+1))$. In the definition, L can be seen as an abstraction of the MXML format and for each task pair in the log, $\Rightarrow$ can be calculated. Subsequently, we can remove less valued pairs by applying a certain threshold value and construct a process model using the remaining pairs.

Figure 6 shows the process model for all cases obtained using the *Heuristics Miner*. Note that the figure is not intended to clearly show all the details of the process model but to provide the reader with a basic idea of the overall process structure. Figure 7 depicts the flow from the ROUTEFEZ to its subsequent tasks, which is one of the tasks in the large process model. In the diagram, boxes correspond to tasks, the numbers within the boxes show how often this task occurred, and the number next to the arc indicates how strong
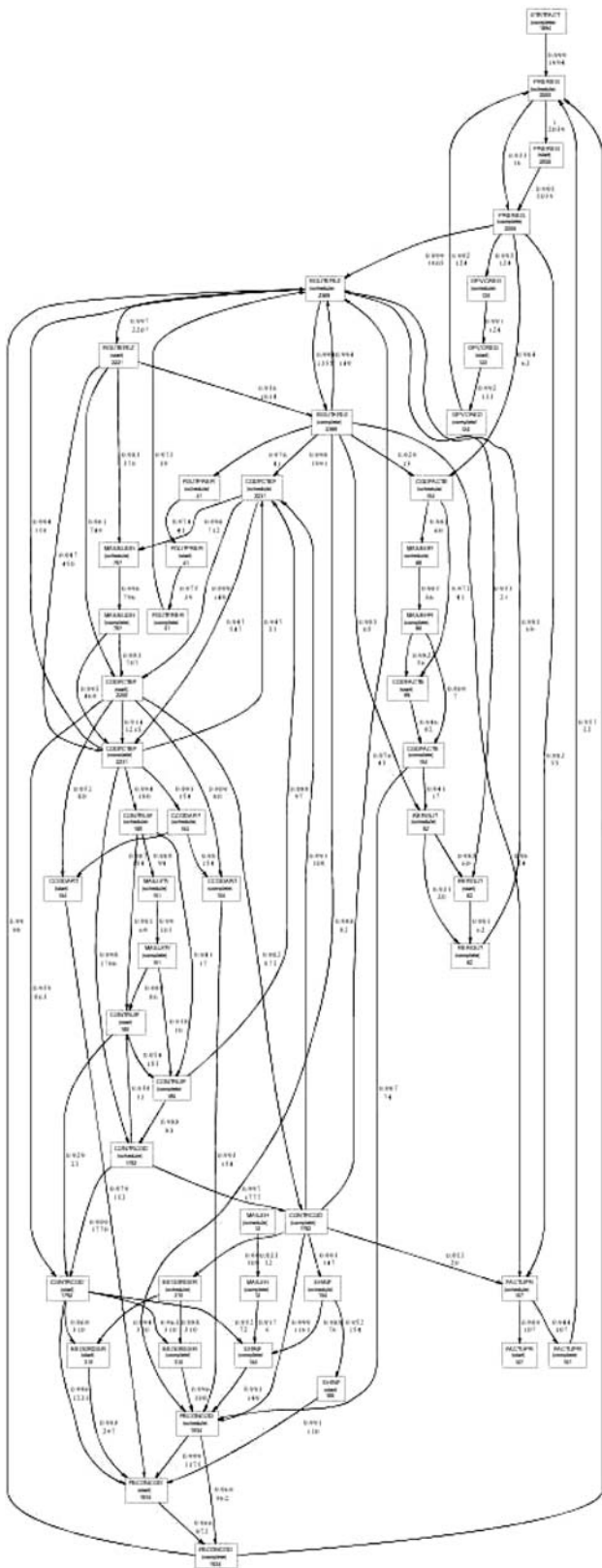
the connection is. We used these diagrams to select the task pairs that were used in our further analysis.
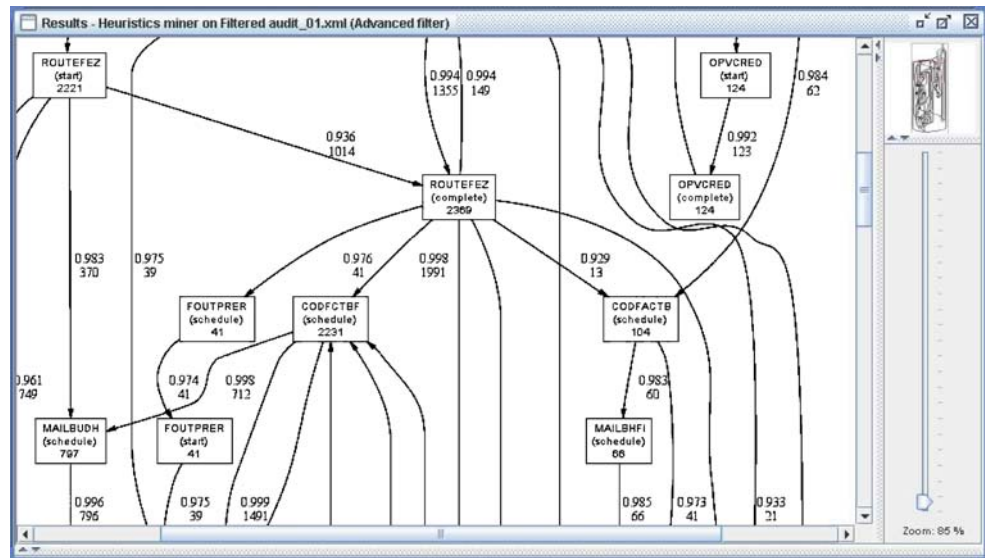
This section described the overall process log and initial analysis results to give an insight for the process logs and its process model. In the next section, we will concentrate on the analysis for answering our research questions.

4.3 Analysis procedure

After investigating the entire process log, we decided to focus our attention to two specific elements. First of all, we decided to analyze the processing times of five specific *tasks*, being the most important checks as prioritized by the central financial management. The five tasks are CODFCTBF, CONTRUIF, ROUTEFEZ, CONTRCOD, and FBCONCOD, as explained in Section 4.1. Note that the five tasks can be performed in several geographical locations. We left out administrative tasks like scanning, keying in data, categorizing, archiving, etc. Secondly, we considered four *pairs* of tasks where we could establish that at times they were subsequently performed within the *same* geographical unit and at other times across *different* units. They are ROUTEFEZ-CODFCTBF, CODFCTBF-CONTRUIF, CODFCTBF-CONTRCOD, and CONTRCOD-BEOORDSR. This choice was made on the basis of our initial analysis, as described in the previous section.

Then, we determined the processing time of each task and the transfer time of each pair of tasks. Before the actual mining started, the process data was filtered to focus on the selected tasks and pairs of tasks. The ProM framework provides several filters that enable the removal of irrelevant information from process logs. Figure 8 shows a ProM screenshot displaying four different filters. The *event log filter* is used to extract the events in which we are interested. In the case of the processing time, the start event and the complete event of the task (e.g. the CODFCTBF start, the CODFCTBF end) are selected, while the complete event of the predecessor and the start event of successor (i.e. the ROUTEFEZ complete, the CODFCTBF start) are selected in the case of the transfer time. The *duplicate task filter* is used to get rid of the duplicate tasks within one instance. In the paper, we took into account only the first task, even though the same task may appear several times in a process instance. The *anonymous log filter* is a customized filter used in this case study. Since the start event in the log contains "system" as an actor, the filter replaces it with an actual actor who is registered in the complete event of the same task. The last log filter is the *replacement filter* which swaps an attribute with another attribute

Fig. 6 Discovered process model from the whole process log

Fig. 7 Zoomed process model that shows ROUTEFEZ task and its subsequent tasks

in the audit trail entry. In this paper, we focused not on process actors but on geographical locations where tasks were performed. So, we applied the replacement filter to substitute a geographical location for an actor.

After applying the filters, we applied the *performance sequence diagram analysis* plug-in. This plug-in makes a sequence diagram from process logs and shows performance measures such as average throughput time, transfer time, time spent in a task, etc. A sequence diagram has vertical and horizontal dimensions.

The vertical dimension is a time dimension and the horizontal dimension shows classifier roles.

Figure 9 depicts the sequence diagram of transfer time of the ROUTEFEZ-CODFCTBF pairs. In the figure, two kinds of patterns can be distinguished: boxes and arrows. When a transfer happens *within* a geographical location, it is represented as a box. If it happens *between* geographical locations, an arrow between them is drawn. As shown in the figure, the transfer times also vary according to geographical locations. For

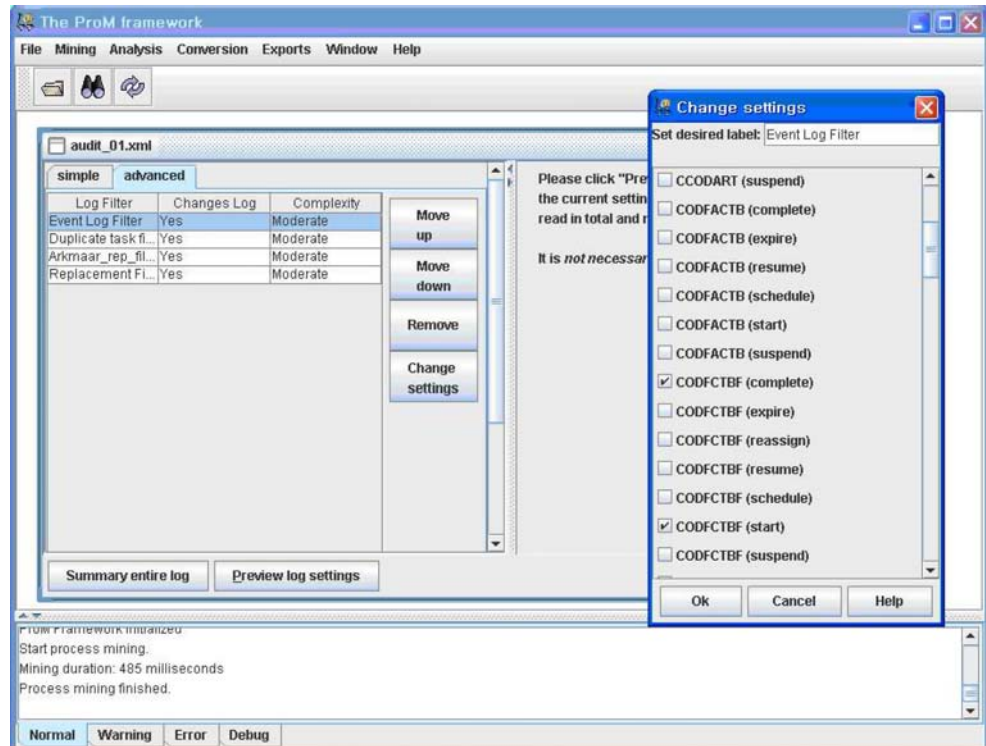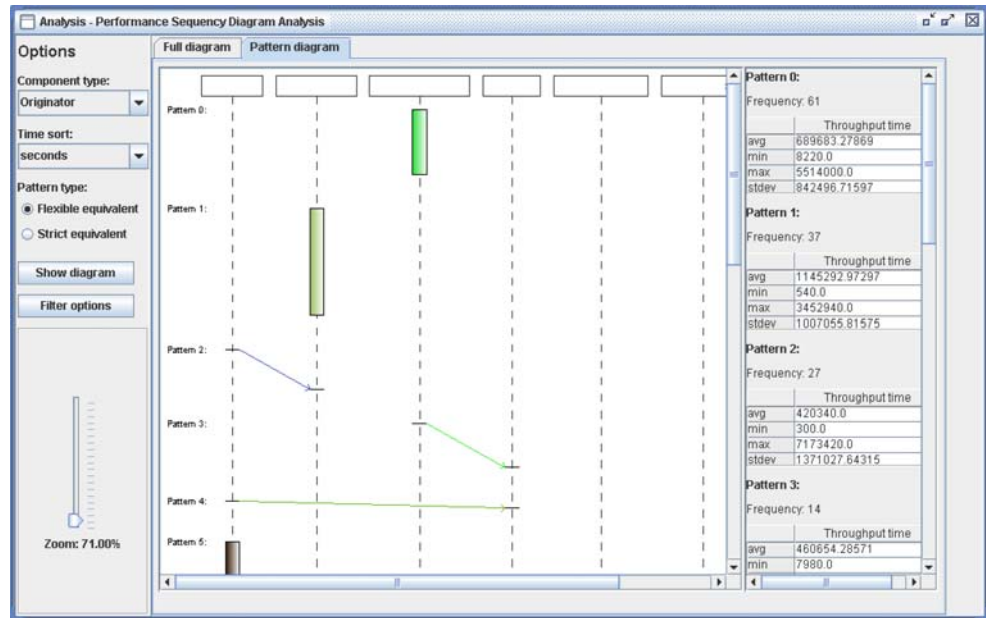**Fig. 8** ProM screenshot showing filters

Fig. 9 ProM screenshot
showing sequence diagram

example, *pattern 0* and *pattern 1* represent a transfer within a geographical location. *Pattern 0* happens within the third location and takes about 8 days, while *pattern 1* takes place within the second location and takes about 13 days. *Pattern 2* and *pattern 3* represent a transfer between geographical locations. *Pattern 2* transfers a task from the first to the second location and spends about 4 days.

4.4 Analysis and findings

Processing time We calculated the average processing times of all five tasks under consideration. The result for the CODFCTBF task, which covers the largest number of different geographical locations, is shown in Fig. 10. The task involves the check on the legitimacy of the invoice by the responsible budget keeper.

The figure reveals that the averages for the COD FCTBF task differ across the various geographical lo-
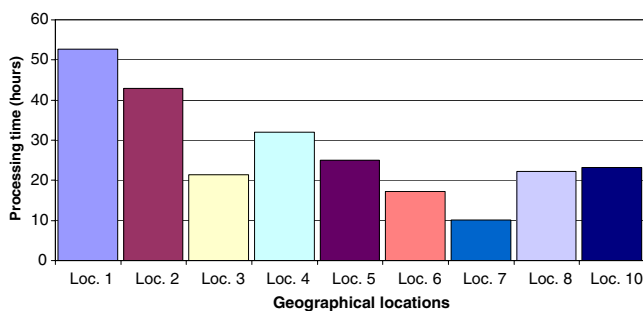

Fig. 10 Average processing time of the CODFCTBF task

cations. These averages range between the extremes of approximately 10 hours and 53 hours. As standardized skewness and kurtosis indicators are within the ranges that may be expected from a normal distribution over the locations, these differences are not extreme. Although the CONTRUIF, ROUTEFEZ, CONTRCOD and FBCONCOD tasks involve fewer geographical locations – respectively only 3, 7, 6 and 2 – the variation is similar to the CODFCTBF task. Because of restrictions of space, we do not show the respective figures.

The result of the Kolmogorov–Smirnov test enables us to reject with a 95% reliability that processing times within any of the tasks are normally distributed. This violates the assumptions for most standard parametric tests to determine statistical differences (e.g. ANOVA), which explains our use of the distribution-free Kruskal–Wallis test that compares medians. For all tasks under consideration, this test leads with a 95% confidence to the outcome that there is a *significant difference between the processing times across various locations.*

To show the relative difference within the processing times for a single task, we present a Box-and-Whisker plot (also known as *boxplot*) for the CODFCTBF task in Fig. 11. In the plot, the medians are shown as notches between the lower and upper quartiles. The plot suggests differences between, for example, the medians of locations 1 and 3, locations 2 and 4, locations 5 and 10, etc.

In order to investigate whether the found differences in processing times across the geographical locations are structural in nature, we divide the overall log into
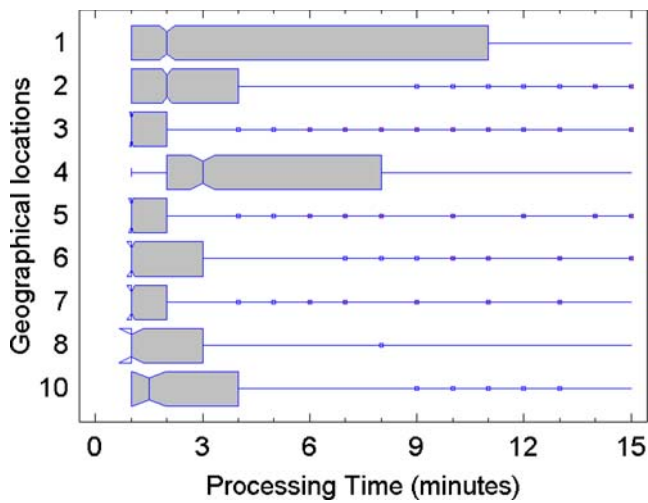
Fig. 11 Box-and-Whisker plot for the CODFCTBF task

6 subsequent smaller logs of equal size and analyzed these as well. As additional analyses confirmed the non-normality of the processing times within all sublogs, we again used the Kruskal-Wallis test. The result is shown in Table 1.

The table shows that for all but the CONTRUIF task the processing times across the locations vary significantly at a 95% confidence level for *all* sublogs. For the CONTRUIF task, this difference is only significant for the 2nd and 3rd sublog and may not be persistent over time. In other words, there may not be a structural difference in effect here. We do reject our first hypothesis as processing times for most of the tasks under consideration tend to differ significantly, and persistently so, across the geographical locations where they are performed.

Transfer time For analyzing the transfer time, we concentrate on the four pairs of tasks that we mentioned before. The transfer points were selected because its involved tasks either take place entirely within the same geographical location or each of the tasks is carried out in a different location. In the first case, we speak of an *intra transfer*, as the work is transferred between executors within the same location; in the

second case, an *inter transfer*, as the executors are at different locations.

For two of the four pairs, there are at most 50 observations of inter transfers versus thousands of observed cases for intra transfers. This does not allow for a meaningful comparison between the different types of transfer. Fortunately, the other two pairs have sufficient data to compare these transfers. Therefore, we focus on the following two pairs:

1. *from ROUTEFEZ to CODFCTBF*: the initial check by a local financial clerk whether an invoice is intended for the sector that the clerk is attached to, and if so, the subsequent check on the legitimacy of the invoice by a budget keeper;
2. *from CODFCTBF to CONTRCOD*: the legitimacy check by a budget keeper followed by the check of a local financial clerk on the control code as filled out by the budget keeper.

For these pairs, there are respectively 2125 and 1764 inter transfers and approximately three times as many intra transfers within each category.

Application of the Kolmogorov–Smirnov test indicates that with a 95% confidence the idea can be rejected that transfer times for either pair are normally distributed. This makes a test that focuses on the comparison of medians of the transfer times more suitable. Figure 12 shows that for both transfer types the median of the inter transfer time exceeds that of the intra transfer time. Note that this difference is the largest in the case of transfers from ROUTEFEZ to CODFCTBF.

Similar as for the analysis of the processing times, the Kruskal–Wallis test was selected to test the equality of medians between intra and inter transfers. In the presence of outliers, Mood's median test was applied as a more robust yet less powerful, additional test. For *both* transfer types, the Kruskal–Wallis test shows significant differences between intra and inter transfers at a 95% confidence interval. At the same confidence level, Mood's median test only shows a significant dif-
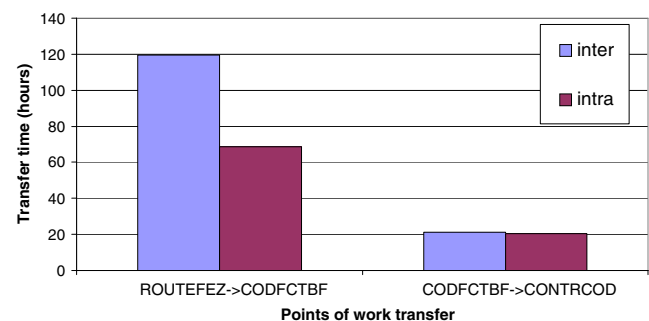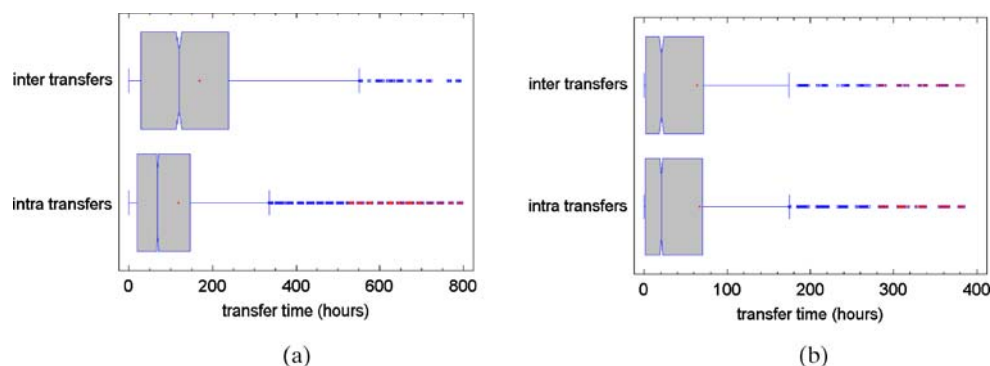
Table 1 The Kruskal-Wallis test result (processing time), significant differences at a 95% confidence interval indicated with '*'

Task	1	2	3	4	5	6
CODFCTBF	*	*	*	*	*	*
CONTRUIF	–	*	*	–	–	–
ROUTEFEZ	*	*	*	*	*	*
CONTRCOD	*	*	*	*	*	*
FBCONCOD	*	*	*	*	*	*

Fig. 12 Median transfer times at transfer points

Fig. 13 Detail
Box-and-Whisker plots
for the transfer of work.
a From ROUTEFEZ to
CODFCTBF. **b** From
CODFCTBF
to CONTRCOD

(a)

(b)

ference for the transfer of work from ROUTEFEZ to CODFCTBF. For both transfer types, box-and-whisker plots that show the area between the lower quartile and upper quartile of the data values with the median in the middle, are given in Fig. 13. Small markers (plus signs) indicate the means for intra and inter transfer times.

So, both statistical tests point at a *significant difference* between the intra and inter transfer times for the transfer of work from ROUTEFEZ to CODFCTBF, where durations of inter transfers clearly exceed those of intra transfer times. The approximate confidence intervals for the medians, indicated by the notches in the quartile bodies in the Box-and-Whisker plot, confirm this result as they are wide apart and do not overlap. The difference is not so apparent for work being transferred from CODFCTBF to CONTRCOD.

Finally, to determine whether the differences between the intra and inter transfer times are of a structural nature, the complete log is split up in 6 subsequent smaller logs of equal size. The procedure is similar as in the case of the analysis of processing times, as described earlier in this section. As Kolmogorov–Smirnov tests confirmed the non-normality of the transfer times within all sublogs, we again used the Kruskal–Wallis test. The result is shown in Table 2.

The table shows that only for the pair ROUTEFEZ-CODFCTBF the significant difference between intra and inter transfers is present in *all* sublogs, hinting at a persistent nature of this difference. Therefore, we reject our second hypothesis, as for this pair at least we see that intra and inter transfer times vary significantly over time.

Table 2 the Kruskal–Wallis test result (transfer time), significant differences at a 95% confidence interval indicated with '*'

Task pair	1	2	3	4	5	6
From ROUTEFEZ to CODFCTBF	*	*	*	*	*	*
From CODFCTBF to CONTRCOD	–	*	*	–	*	*

5 Discussion

5.1 Summary of findings

We rejected our hypotheses that suppose that workflow technology takes away geographical barriers (Hypotheses 1 and 2). With respect to the first part, our analysis shows significant differences between processing times of equivalent tasks across different geographical locations; significant differences between intra and inter transfer times are also found.

5.2 Evaluation

We gathered feedback on the found results from a team of the involved municipality, which included the financial manager, functional administrator of the workflow system, a systems integrator responsible for technical modifications, and a budget-keeper/executor. We had a one-and-a-half hour meeting with them in the city town-hall, where we presented and discussed the results, followed-up by several e-mail contacts and phone conversations.

No satisfactory explanation could be found for the surprising differences in processing times (Hypothesis 1), as the team members once more confirmed that the tasks are intended to be strictly equivalent across the various locations. Differences in local skills and perhaps informal norms may contribute to the difference. This may be in line with research in the tradition of social ecology (Barker 1968). It positions that different social settings, such as offices and meeting rooms, are associated with different behavioral norms, mental schemas, and even scripts that sharply affect the way people act and the expectations they have of others.

After considerable deliberation, a likely explanation was found for the difference in transfer times (Hypothesis 2). Within the municipality, local financial clerks

are provided with reports on "open" invoices. These can be used to urge budget keepers to check the invoices that are with them for some time. The team from the municipality suspects that this encouragement is done more frequently and more persuasively in settings where the clerks and budget keepers are in the same location, which may well explain the distinctive difference between transfer times from ROUTEFEZ and CODFCTBF.

But even if the encounters between financial clerks and budget keepers are not explicitly planned, the effect of spontaneous communication between them are perhaps better not underestimated. After all, it is logical that spontaneous encounters will take place more frequently when people reside in the same building. With spontaneous casual communication, people can learn, informally, how one anothers work is going, anticipate each others strengths and failings, monitor group progress, coordinate their actions, do favors for one another, and come to the rescue at the last minute when things go wrong (Davenport 1994). But, as has been pointed out in other contexts before (Kraut et al. 2002), physical separation drastically reduces the likelihood of voluntary work collaboration.

5.3 Limitations

Clearly, this study is carried out within the setting of a *single* organization, so the usual limitations apply with respect to generalizing its results. A particular difficulty in gathering more general support for our findings will be to find a setting for replicating our case study. In our opinion, it is a rare phenomenon in business that actors involved in a workflow process within the *same* cultural and organizational bounds can be at *different* geographical locations, while the steps they perform are exactly the *same* from a business perspective. Also, because of the nature of the informal interactions that we suggested to be of influence in our case, it will be near to impossible to emulate such a situation in a laboratory setting. These factors also influence the falsifiability of our findings.

A more specific concern could be raised about the validity of reasoning over process performance, as we strongly focused on the analysis of an automatically generated process log. Obviously, process logs are by no means a full representation of what is going on in an organization. However, for the reported case it seems likely that the recorded events follow actual work execution quite closely, as confirmed by the team of the municipality. In another part of the larger research project we are involved in (see Reijers and van der Aalst 2005), we have seen an implementation

where people worked around the workflow system on a wide scale, e.g. using the workflow system in batch mode to check out work that was completed manually much earlier. In such a case, it would be much more dubious to draw conclusions of the kind we did. The patterns in the even logs that hinted at such anomalous behavior, i.e., (1) extremely short processing times and (2) many "bursts" of task completions followed by relatively long periods of inactivity, were not present in the analyzed log.

A final limitation that needs to be mentioned is that only a restricted period (half a year) was used as a time window for the evaluation of the invoice handling's process performance. We attempted to counter this issue with carrying out our analyses on the level of sub-logs as well, but we cannot rule out entirely that we have witnessed a temporal effect. Yet, team members of the municipality could not indicate factors that made this period distinctly different from other periods of operation.

5.4 Implications for practical use

This study shows that geographical proximity of workers favors their interaction, as was already suggested by the work of Allen (1977). The most important implication for practice is that workflow technology should not be assumed to level all geographical barriers between people just by itself. While effective to coordinate work between distributed actors, workflow technology cannot replace the communication patterns that arise relatively easily between people who are close to each other.

As a spin-off from this insight, organizations may want to reconsider the assignment procedures applied by their WfMSs to improve process performance. If there is freedom of choice between several available agents for executing a work step for a particular case, it seems sensible that the resource is preferred that is geographically closest to the previous agents that have been involved in handling that particular case. This heuristic would be a variation of the "case assignment" best practice, as described in Reijers and Liman Mansar (2005). The expected effect will be that chances increase that communication between agents that are geographically close helps to improve the smooth execution of cases they are jointly responsible for.

On a more general level, our study may be interpreted as a caution against hasty decisions to outsource parts of business processes, as mostly motivated from expected efficiency gains. It is a well-known insight that quality of communication is a major determinant in trust and business understanding in outsourcing

contexts (Lee and Kim 1999). Also, it has been noted in the context of virtual teams that under circumstances of multiple cultures and lack of a common language, consistent communication becomes even more important, especially given distributed constraints, e.g., general inability to meet face to face or even at the same time (Qureshi et al. 2006). So, if geographical distance between the actors from our study within the same *municipality* has such a profound effect on process performance, what would that imply for the performance of actors in a collaborative process distributed over a geographically much larger scale? Assuming all other circumstances equal, it seems difficult to argue that effects will be less pronounced that we encountered.

6 Conclusions

WfMSs are supposed to efficiently and effectively support actors in the collaborative business processes they are involved in *regardless* of their geographical location. In this paper, we critically evaluated this idea through a study into the performance of a WfMS in a concrete organizational context. We analyzed a large workflow process log with the ProM framework and its associated tools and found that the geographical location of actors and the distance between them were major distinguishing factors in process performance. The feedback from the organization brought a partial explanation for this phenomenon: People are more inclined to urge others to complete their work when they are geographically close to each other. Also, the positive effects of spontaneous interactions between collocated workers may be at work here.

Our paper contributes to a better understanding of collaboration processes that are distributed over various geographical locations. Its outcomes are at odds with a widely held belief that technology is effective and sufficient to have distributed performers work together *without* repercussions on the effectiveness of the process as a whole. A more specific contribution is that our work adds to the only marginal collection of empirical, quantitative evaluations of workflow management technology. Finally, the analysis in this paper once more demonstrates the feasibility of process mining techniques in evaluating current situations or answering managerial questions related to process enactment. The interested reader is referred to our earlier work in this field, e.g. in the setting of a public works department (van der Aalst et al. 2007a).

In future work, we plan to repeat our analysis with logs from other organizations, taking into account other potential factors affecting performance (e.g. organizational hierarchy). It would be highly desirable to find organizations with highly distributed actors for reasons of comparison and we invite them to contact us to cooperate. Also, we are currently finalizing the longitudinal study we mentioned in the introduction in this paper to come up with a broader perspective on performance gains (or losses) through the application of workflow technology.

Another, much more ambitious line of research would extend to the analysis and design of those local conditions that favor a desirable performance pattern in a distributed collaborative process. Clearly, this is a many-faceted research line, where insights from the social ecology tradition (Barker 1968), experiences with virtual teams (Maznevski and Chudoba 2000), and the influences of physical space on human interaction (Allen 1977) are only some of the relevant starting points that can be imagined.

References

Allen, T. (1977). *Managing the flow of technology: Technology transfer and the dissemination of technological information within the R&D organisation*. Cambridge, MA: MIT Press.
Barker, R. G. (1968). *Ecological psychology*. Stanford, CA: Stanford University Press.
Becker, J., & Vossen, G. (1996). *Geschaftsprozesmodellierung und workflow-management: Eine einfuhring*, chapter Geschaftsprozesmodellierung und workflow-management (pp. 17–26). Albany: Modelle, Methoden, Werkzeuge.
Blake, M. B., & Huhns, M. N. (2008). Web-scale workflow: Integrating distributed services. *IEEE Internet Computing Magazine, 12*(1), 55–59.
Boutellier, R., Gassmann, O., Macho, H., & Roux, M. (1998). Management of dispersed R&D teams. *R&D Management, 28*(1), 13–25.
Carmel, E., & Agarwal, R. (2001). Tactical approaches for alleviating distance in global software development. *Software, IEEE, 18*(2), 22–29.

Davenport, T. H. (1994). *Process innovation*. Boston, MA: Harvard Business School Press.

Fakas, G. J., & Karakostas, B. (2004). A peer to peer (P2P) architecture for dynamic workflow management. *Information and Software Technology, 46*(6), 423–431.

Grasso, A., Meunier, J. L., Pagani, D., & Pareschi, R. (1997). Distributed coordination and workflow on the world wide web. *Computer Supported Cooperative Work* (*CSCW*)*, 6*(2), 175–200.

Grefen, P., Ludwig, H., & Angelov, S. (2003). A three-level framework for process and data management of complex e-services. *International Journal of Cooperative Information Systems, 12*(4), 487–531.

Herbst, J., & Karagiannis, D. (2004). Workflow mining with InWoLvE. *Computers in Industry, 53*(3), 245–264.

Herrmann, T., & Hoffmann, M. (2005). The metamorphoses of workflow projects in their early stages. *Computer Supported Cooperative Work* (*CSCW*)*, 14*(5), 399–432.

IDS Scheer (2002). ARIS process performance manager (ARIS PPM). http://www.ids-scheer.com.

Kraut, R. E., Fussell, S. R., Brennan, S. E., & Siegel, J. (2002). Understanding effects of proximity on collaboration: Implications for technologies to support remote collaborative work. In P. Hinds & S. Kiesler (Eds.), *Distributed work* (pp. 137–162). Cambridge, MA: MIT Press.

Kueng, P. (2000). The effects of workflow systems on organizations: A qualitative study. In W. M. P. van der Aalst, J. Desel, & A. Oberweis (Eds.), *Business process management: Models, techniques, and empirical studies* (pp. 301–316). Berlin: Springer.

Lawrence, P. (Ed.) (1997). *Workflow handbook 1997, workflow management coalition*. New York: Wiley.

Lee, J. N., & Kim, Y. G. (1999). Effect of partnership quality on IS outsourcing success: Conceptual framework and empirical validation. *Journal of Management Information Systems, 15*(4), 29–61.

Leymann, F., & Roller, D. (2000). *Production workflow: Concepts and techniques*. New Jersey, USA: Prentice Hall PTR, Upper Saddle River.

Maznevski, M. L., & Chudoba, K. M. (2000). Bridging space over time: Global virtual team dynamics and effectiveness. *Organization Science, 11*(5), 473–492.

Oba, M., Onada, S., & Komoda, N. (2000). Evaluating the quantitative effects of workflow systems based on real cases. In *Proceedings of the 33rd Hawaii international conference on system sciences* (Vol. 6, pp. 6031). Washington DC: IEEE.

Palmer, N. (2007). A survey of business process initiatives. BPTrends report, Business Process Trends. http://www.bptrends.com.

Qureshi, S., Liu, M., & Vogel, D. (2006). The effects of electronic collaboration in distributed project management. *Group Decision and Negotiation, 15*(1), 55–75.

Reijers, H. A., & Liman Mansar, S. (2005). Best practices in business process redesign: An overview and qualitative evaluation of successful redesign heuristics. *Omega, 33*(4), 283–306.

Reijers, H. A., & van der Aalst, W. M. P. (2005). The effectivenes of workflow management systems: Predictions and lessons learned. *International Journal of Information Management, 56*(5), 457–471.

Rinderle, S., Reichert, M., & Dadam, P. (2004). Flexible support of team processes by adaptive workflow systems. *Distributed and Parallel Databases, 16*(1), 91–116.

Sengupta, K., & Zhao, J. L. (1998). Improving the communicational effectiveness of virtual organizations through workflow automation. *International Journal of Electronic Commerce, 3*(1), 49–69.

Staffware (2000). Benefits, progress and competitive edge: The customer's experience. Maidenhead: Vendor publication, Staffware.

Steinfield, C., Jang, C. Y., & Pfaff, B. (1999). Supporting virtual team collaboration: The TeamSCOPE system. In *Proceedings of the international ACM SIGGROUP conference on supporting group work* (pp. 81–90). New York, NY, USA: ACM.

Townsend, A. M., DeMarie, S. M., & Hendrickson, A. R. (2000). Virtual teams: Technology and the workplace of the future. *IEEE Engineering Management Review, 28*(2), 69–80.

van der Aalst, W. M. P., Reijers, H. A., & Song, M. (2005). Discovering social networks from event logs. *Computer Supported Cooperative Work, 14*(6), 549–593.

van der Aalst, W. M. P., Reijers, H. A., Weijters, A. J. M. M., van Dongen, B. F., de Medeiros, A. K. A., Song, M., et al. (2007a). Business process mining: An industrial application. *Information Systems, 32*(5), 713–732.

van der Aalst, W. M. P., Weijters, A. J. M. M., & Maruster, L. (2004). Workflow mining: Discovering process models from event logs. *IEEE Transactions on Knowledge and Data Engineering, 16*(9), 1128–1142.

van der Aalst, W. M. P., van Dongen, B. F., Günther, C. W., Mans, R. S., de Medeiros, A. K. A., Rozinat, A., et al. (2007b). ProM 4.0: Comprehensive support for real process analysis. In M. van Eekelen, P. Groot, & A. Serebrenik (Eds.), *26th international conference on applications and theory of Petri nets* (*ICATPN 2007*). Berlin: Springer.

van der Aalst, W. M. P., & van Hee, K. M. (2002). *Workflow management: Models, methods, and systems*. Cambridge, MA: MIT Press.

van Dongen, B. F., Busi, N., Pinnaand, G. M., & van der Aalst, W. M. P. (2007). An iterative algorithm for applying the theory of regions in process mining. In K. van Hee, W. Reisig, & K. Wolf (Eds.), *Proceedings of the workshop on formal approaches to business processes and web services* (*FABPWS'07*). Siedlce: Publishing House of University of Podlasie.

van Dongen, B. F., & van der Aalst, W. M. P. (2004). EMiT: A process mining tool. In *Applications and theory of Petri nets 2004, 25th international conference, ICATPN 2004, Bologna, Italy, 21–25 June 2004, Proceedings. Lecture notes in computer science* (Vol. 3099, pp. 454–463). Berlin: Springer.

Vonk, J., & Grefen, P. (2003). Cross-organizational transaction support for e-services in virtual enterprises. *Distributed and Parallel Databases, 14*(2), 137–172.

Weijters, A. J. M. M., & van der Aalst, W. M. P. (2003). Rediscovering workflow models from event-based data using little thumb. *Integrated Computer-Aided Engineering, 10*(2), 151–162.

Hajo Reijers is an assistant professor of business process management at Eindhoven University of Technology. He received his PhD and MSc in computer science from the same institute and has worked for several management consultancy firms, most recently as a manager within Deloitte. His research interests include business process redesign, workflow management systems, and discrete event simulation. He published articles in Information Systems, Journal of Management Information Systems, Computer Supported Cooperative Work, Omega: The International Journal of Management Science, Computers in Industry, and other journals.

Minseok Song is a postdoctoral researcher at the Information Systems department of the Faculty of Technology Management of Eindhoven University of Technology. He received his PhD degree in department of Industrial & Management Engineering at Pohang University of Science and Technology in 2006. His research interests include workflow, business process management, process mining, process knowledge management, simulation, and social network analysis. He published articles in Information Systems, Computer Supported Cooperative Work, Computers in Industry, International Journal of Computer Integrated Manufacturing, and other journals.

Byungduk Jeong received the B.S. in Industrial Engineering department at KAIST in 2005. He is currently a Ph.D. student in Industrial Engineering department at KAIST. His research area includes systems biology, bio CAD and general CAGD problems. He is also interested in business process management related topics.