Antunes, P., Baloian, N., Zurita, G., & Pino, J. (2018). Supporting People-Driven, Dynamic and Geo-Located Work Processes. In C. Stary (Ed.). 10th International Conference on Subject-Oriented Business Process Management, Linz, Austria, April 5-6, 2018 (S-BPM ONE) (pp. 137-146). New York, NY: ACM.

Supporting People-Driven, Dynamic and Geo-Located Work Processes

Pedro Antunes
Victoria University of Wellington
P.O. Box 600, Wellington 6140
New Zealand
pedro.antunes@vuw.ac.nz

Nelson Baloian
Computer Science Department (DCC)
Universidad de Chile
Chile
nbaloian@dcc.uchile.cl

Gustavo Zurita
Dept. of Management Control and Information System. Economics and Business Faculty. Universidad de Chile
gzurita@fen.uchile.cl

José A. Pino
Computer Science Department (DCC)
Universidad de Chile
Chile
jpino@dcc.uchile.cl

ABSTRACT

Some work scenarios foster the adoption of people-driven, dynamic and geo-located processes. To support such scenarios, we suggest two fundamental changes in process structure and control. Regarding structure, we move away from traditional process models towards process contexts, which can be organized around geographical locations. Regarding control, we move away from model-based control-flow towards dynamic activities defined by the participants as processes unfold. This research makes the following unique contributions: 1) It provides the first implementation of people-driven dynamic processes; 2) It provides the first implementation combining people-driven dynamic processes and geographical context; 3) Finally, it provides a unique approach to build process context, which leverages the possibilities brought by microblogging platforms in exchanging semi-structured and unstructured messages.

CCS CONCEPTS

• **Applied Computing** → **Enterprise computing**; *Business process management*

KEYWORDS

Dynamic definition of activities; Process context; Contextually-supported human control; Twitter use.

ACM Reference format:

P. Antunes, N. Baloian, G. Zurita, J. A. Pino., 2018. Supporting People-Driven, Dynamic and Geo-Located Work Processes. In *Proceedings of S-BPM ONE 2018, Linz Austria, April 2018 (S-BPM ONE 2018)*, 10 pages.

1 INTRODUCTION

People-driven dynamic processes are those in which the activities and their execution order are determined as the process unfolds, depending on the interactions and decisions made by human workers [1]. This type of process is particularly adequate to scenarios where some parts of the work either cannot be foreseen or are left open for people to decide. Many of these scenarios also involve the geographical context, as illustrated below.

Consider for instance, occasional rubbish collection. Since orders from clients are defined on a daily basis, the collection process cannot be completely planned. However, it also cannot be completely geo-referenced or simply ad hoc. Some degree of control-flow is necessary to ensure quality of service and adequate performance. Maintenance work done by electricity distribution companies follows the same pattern. Even though the maintenance activities can be carefully planned and optimized, many events may occur during the day that prevent a process to evolve as expected. For instance, clients may not be on site to give access to contractors. A request to cut distribution may be suddenly cancelled because the client found a payment receipt. And of course, urgent repairs may overtake other planned activities. Another scenario considers firefighting. To avoid uncertainty, firefighting is based on extensive training, experienced professionals, and multiple contingency plans. However, each fire is a unique case. Depending on how it evolves, firefighters may be forced to dynamically change the command structure, responsibilities, goals, coordination protocols, etc. [39].

All above scenarios involve humans in making dynamic changes in process execution, as the work context changes over time. Furthermore, geographical context also plays an important role in determining the process execution, since activities are location-dependent and the participants are constantly on the move to accomplish their work.

The adequate support to these scenarios has generated considerable interest in the BPM (Business Process Management) community regarding their implications to process modelling, enactment, management, and execution. Various alternatives have been studied, including how to increase the flexibility of model-based execution [2], how to handle unique cases [3], how to model people-driven activities [4], and also how to integrate geographical context in process models [5]. However, a more recent concern is how to democratize BPM [6] by empowering the process participants to determine the process evolution at their own discretion, while at the same time preserving some important characteristics of the BPM approach such as control, visibility and traceability.

So far, such democratization of BPM has been mainly addressed in a conceptual way [4, 6]. In this paper, we discuss the implementation of a BPM system that brings together the following features:

- Support *people-centered* process enactment and execution, where processes can be enacted and activities can be defined by the process participants without any model-based constraints;
- Support *dynamic* process execution, where activities can be defined and changed as the process unfolds;
- Support *geo-location* as a complementary way to structure process execution.

The system has been developed using the example scenarios already discussed. In all these scenarios, events, locations, emergent needs, and other contextual factors may lead human workers to override model-based control-flow (e.g. B can only be executed after A finishes). Even though process execution may be foreseen, it cannot be statically defined: it must be dynamic, people-driven and context aware.

The research reported in [7] discussed a set of requirements for integrating geo-located activities with BPM, highlighting in particular the conflict between spatial and task dependencies in control-flow. The authors suggested that the conflict could be solved by giving predominance to spatial dependencies. This paper presents a concrete implementation of that proposition plus some new ideas. The new ideas emphasize the people-driven approach, which centers process enactment, management and execution on the process participants.

Our implementation provides a graphical, easy to use interface, which combines geographical and process visualization. It also implements a people-driven solution for efficient assignment of activities in which workers self-manage the process execution.

The next section discusses previous work that has been done in this area. Section 3 discusses in more detail the system features. Section 4 describes the implemented system, and section 5 presents results from a preliminary, formative evaluation. Section 6 concludes the paper.

2 PREVIOUS WORK

The BPM approach is very attractive for structuring most types of work in organizations. The work activities are usually presented in a descriptive model that is typically simple, understandable and elegant. Furthermore, these descriptive models can be used to control the process execution (what we designate by control-flow), usually by process aware information systems (PAIS) [8]. The coordination of activities is normally predetermined by analysts at design time and controlled by PAIS at runtime. Process participants then interact with worklist handlers to execute the activities assigned to them.

However, for long it has been recognized that model-based execution faces many challenges, especially in organizations needing some degree of flexibility [2]. A large body of research has been devoted to this problem, which because of its complexity cannot be detailed in this paper (we recommend [9] and [10] for a more comprehensive overview). Nevertheless, we can summarize the existing viewpoints and approaches to address the problem.

Well-structured processes with ad hoc activities. This approach regards processes as essentially well-structured and model-based. However, sometimes variations and exceptions occur, which require additional rules supporting ad hoc interventions. Two well-known solutions have been proposed: exception handling and flexible BPM. The former integrates rules and mechanisms to handle expected and unexpected events, typically suspending or cancelling a process to execute an alternative handling procedure [2, 11, 12]. The latter brings constraints to process models, thus allowing alternative activities and flows, provided they do not violate the specified constraints [13, 14, 40]. All in all, the two approaches support a degree of improvisation, process variants and ad hoc activities, however under the scope of model-based process execution.

Event-driven processes. Event-driven BPM suggests that we should model process events instead of activities [15, 16]. Since activities are not defined, the process participants effectively have ample latitude to perform the activities in their own ways. Nevertheless, process execution is still model-based.

Adaptive processes. An approach designated adaptive process modelling suggests the use of process fragments in process models [17]. Process fragments define regions where optional models can be dynamically selected to execute work in different ways. This approach does not mean we can model people-driven dynamic work, but instead that we can define regions where work may be changed depending on contextual factors. However, outside these regions, work is still confined to model-based execution.

Semi-structured processes with collaborative activities. This approach combines the typical model-based BPM with social media, a concept often designated as social BPM [18, 19]. The shared space provided by social media can be used to assign activities to people in a participative way. This approach definitely differs from the previous ones by emphasizing the role of people in process execution. It can therefore be seen as a first, albeit limited approach to people-driven dynamic processes.

Dynamic subject-oriented processes. Social Business Process Management (S-BPM) has recently gained attention as a BPM alternative which emphasizes how humans participate and collaborate within the process scope, instead of just seeing humans as actors designated to execute specific activities [6]. Taking this viewpoint, researchers started to investigate how to model dynamic behavior using particular model constructs [20]. Researchers have also started to investigate how to support these dynamic processes [4, 21]. Even though promising, we have not seen an actual implementation of these concepts.

Knowledge-intensive people-driven processes. The main focus of this approach is on the people driving process execution. Its initial impetus was brought by the concept of case management. In opposition to traditional BPM, which concerns repetitive, systematic work, case management deals with unique and knowledge-intensive work. With case management, control is moved from activities to a case file [22, 23]. Two more specific approaches can be found in this category: adaptive case management [3, 24] and emergent case management [25]. The former adds to-do lists to case files, which can be seen as an equivalent to process models but without control-flow [24]. The latter emphasizes the collaborative management of case files, using in particular social media and microblogging for communication. When compared to the previous approaches, case management represents a radically different way to handle work, as it becomes fully unstructured. However, it seems too radical for the work scenarios discussed in this paper: case management is clearly centered on the specific case of highly-skilled workers performing knowledge-intensive work.

Interaction-intensive people-driven processes. In this approach, we also find a significant concern with the human involvement in the process dynamics [26-28]. However, instead of giving primacy to knowledge-intensive work, the emphasis is on human interaction [20]. One innovative solution that has been suggested to manage the execution of unstructured work consists in using machine learning to automatically identify dependencies between activities and to suggest who should execute a certain activity [26, 27].

In summary, we observe that dynamic process execution has been addressed with a range of solutions that extend from the model-driven to the people-driven. The former case emphasizes model-based control, which in special circumstances can accommodate ad hoc activities, while the latter emphasizes human discretion in determining what to do next.

The system described in this paper adopts the latter viewpoint with a specific focus on process execution, which seems to be a current gap in the research literature. Even though researchers already equated how to integrate people-driven dynamic processes in process modelling, from a conceptual perspective, in this research we are concerned with process execution.

Since our system integrates geo-location with process execution, it also seems relevant to briefly overview research in that area. However, little work has been done so far. [29] developed a framework for integrating visualization applets into worklist handlers, which allows workers to select work items taking geo-location into consideration. [30] adopts a similar approach, integrating GIS tools with worklist handlers. However, these two studies do not actually address the broader problem of integrating geo-location into process management, as they only concern worklist handing. [5, 31, 32] extended BPM modelling to include location-dependencies in control-flow, e.g. using location-dependent parallel splits and synchronizations, but they did not address the use of geo-location in dynamic processes. To the best of our knowledge, our research is unique in integrating geo-location into people-driven dynamic processes beyond worklist handing.

3 FEATURES

We now briefly discuss some key features of the developed system.

Process structure. Process structure concerns the way in which activities are pulled together in a process. The most common approach to process structure is to define process models at design time, which define the sequences in which activities take place during execution [33]. (However, other approaches exist, e.g. based on visual narrative [34-36].) Such models are then used by PAIS to enact, execute and manage the processes [37]. However, as discussed in Section 2, this approach constrains dynamic changes during process execution. In order to avoid this constraint, we decided to include geo-location as an alternative process structure. Prior research [38] into dynamic work scenarios highlighted that spatial data provide an adequate frame for situation awareness and action, which can be used to structure work. Therefore, the first feature considers that:

> *People-driven, dynamic, and geo-located processes are primarily structured by geographical locations.*

As with the interaction-intensive people-driven approach discussed in Section 2, which moved process structure away from models towards communication, or the knowledge-intensive people-driven approach, which moved process structure to case files, in our approach we move process structure away from models towards geographical locations.

Process control. Process control concerns how the activities defined by a process are actually managed during execution time. The most common approach is to use model-based control-flow, where the process model is used to determine how the sequence of activities is executed (However, other approaches exist, e.g. adaptive case management [24].) It derives from this viewpoint that PAIS are effectively in control of the process. However, as already discussed in Section 2, this approach reduces the involvement of humans in making dynamic changes to processes, while they are being executed. In order to avoid this constraint, we decided to relinquish control from PAIS and instead give that control to humans. Likewise the knowledge-intensive people-driven approach, processes become unframed [37]. The decisions on how to execute activities will then depend on human communication and determination. Therefore, the second feature we consider is:

> *People-driven, dynamic, and geo-located processes are primarily controlled by the process participants; information systems will only have a supportive role and will not constrain any dynamic changes required by the process participants.*

Unlike the knowledge-intensive people-driven approach, which adopted the collaborative management of case files, we propose combining process communication with process context.

Process communication. In order to control the process, the process participants will have to communicate more (than with the model-based approach). In particular, semi-structured messages have to be exchanged to enact processes, initiate activities, request and pass control over activities, notify that an activity has been completed, etc. Unstructured messages are also necessary, e.g. to discuss who can take responsibility for an activity, who could substitute a participant unable to complete an activity, or even what should be done in case of inappropriate behavior. We suggest such communication can be supported by microblogging:

> *People-driven, dynamic, and geo-located processes can be supported by microblogging, which provides a conduit for exchanging semi-structured and unstructured messages about a process; process participants rely on these messages to dynamically manage processes.*

Process context. Since we move control from PAIS to the process participants, the process participants need to access the process context to decide which activities to take, and when and where to executed them. Furthermore, since activities are dynamically executed, the process context must also track the process enactment and evolution:

> *People-driven, dynamic, and geo-located processes are supported by context, which tracks where, who and what activities have been done.*

In a way, in the scope of dynamic processes, we can regard the process context as an alternative to the process model. Unlike the knowledge-intensive people-driven approach, which is centered on the case file, our approach is centered on the process context, which includes geographical locations. Additionally, the process context supports visibility and traceability of process execution.

4 PEOPLE-DRIVEN, DYNAMIC AND GEO-LOCATED PROCESS MANAGEMENT SYSTEM

A prototype was developed to explore the feasibility of implementing a system which complies with the features discussed in the previous section. For this reason, we decided to implement a system having only the most important functionalities we thought were necessary for that. Therefore, we consider only one human role in the system, which is able to enact processes, dynamically add and modify activities, pick and execute activities, and communicate with other process participants. Next, we describe in detail the system functionality and discuss in particular how the system uses the Twitter platform for process communication.

4.1 Process enactment

Users login into the system using their Twitter credentials. When logging in for the first login, they can declare their expertise (roles), so the system can push activities that best fit their profile. A user enacts a process by giving it a name. This is the only required information at the beginning, because the rest can be dynamically specified during execution. In particular, activities and flows can be re/defined at any time after enactment. It is also possible to add an activity to a process at any time, and existing activities can also be modified. Fig. 1 shows the main view of the user interface during process enactment. This functionality fosters open participation in the process, which in turn depends on personal responsibility rather than machine control to execute the process.

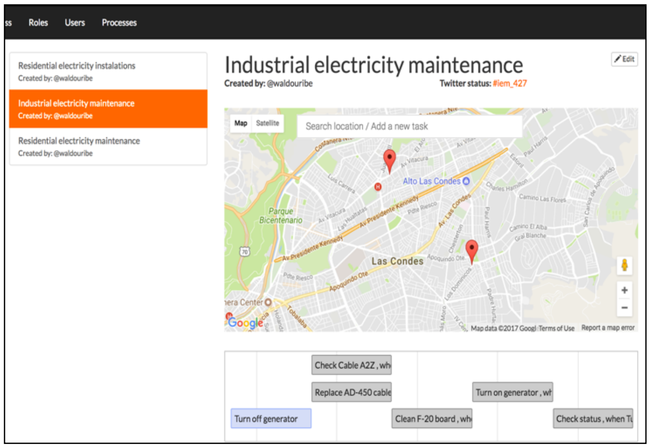


Figure 1: Process enactment: a list of enacted processes is shown on the upper left part of the window. The timeline at the bottom-right shows the so far defined activities for a selected process. The map shows the locations of activities defined for a selected process (if location information is defined).

According to the decision to give predominance to the spatial context of activities, a map showing the location of defined activities takes a prominent part of the user interface. Along with this, the user interface also shows the list of enacted processes and a timeline with activities. Fig. 1 shows a process enacted with the name "Industrial Electricity Maintenance"; it was created by user @waldouribe with the hashtag #item_427, which follows Twitter conventions. The timeline shows six activities. The first one, named "Turn off the generator", appears slightly brighter than the others, indicating that it is the only one that can currently be executed. We see that two other activities can be executed in parallel, but only after the first one is completed. When an activity is completed, it is shown in a dark color.

After enacting a process, the following actions can be done:

- **Defining roles**: a role corresponds to a specific set of skills that may be required to execute an activity. Fig. 2 shows the user interface that can be used to define roles for a process. Roles are part of the process context. Since we allow to dynamically change the process, roles are considered as part of context rather than core process information. Furthermore, roles can be created either before or after activities have been defined.

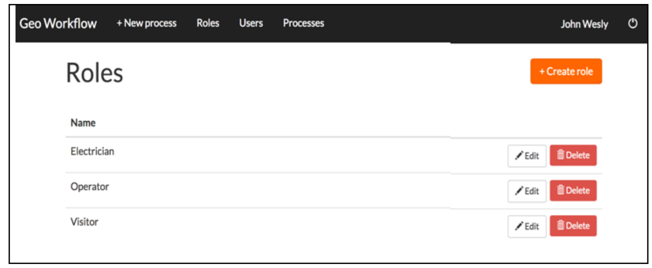


Figure 2: Defining roles for activities.

- **Creating an activity**: activities can be created by clicking a location on the map or the timeline. When using the map, a pin will appear showing the place where the activity should be executed. Activities always appear in the timeline.
- **Defining activity attributes**: By selecting an activity in the timeline or map, the following attributes can be defined and/or changed:
  - **Description**: brief description of what should be done.
  - **Priority**: a number between 1 and 10, which suggests the importance of this activity. This information is intended to help users making decisions about which activities to pick first.
  - **Dependencies:** this is a (possibly empty) list containing names of other activities which are required to be completed before this one can start. This information is taken in order to compute the right place for the activity in the timeline. All activities with no dependencies are displayed in parallel at the beginning of the timeline. The system checks for possible inconsistencies in the definition of dependencies, like circular references and deadlocks.
  - **Roles:** this is a (possibly empty) list mentioning the roles that workers should have to pick and execute an activity. When more than one role is defined, the system assumes any worker having at least one of the listed roles can perform the activity. When no role is specified, any worker can perform the activity.
  - **Starting and ending times:** these attributes are not set by users, because they correspond to the times when the activity was actually executed. These attributes are set by the system.

Fig. 3 shows the user interface when changing the attributes of an activity. It illustrates the definition of attributes for activity “Clean F-20 board”. This activity has priority 3. The activity should be performed by an electrician. This activity can also be defined as dependent on other activities defined for this process: “Turn off generator” and “Replace T-450 cable”. In the darkened background, we see that the current activity can only start after “Replace T-450 cable” has been completed. We also see that the disposition of the bars representing activities in the timeline shows that “Replace T-450 cable” can start only after “Turn generator off” is completed. We also see that there is a fourth activity defined with the name “Turn generator on”, which is displayed after the current one in the timeline, meaning it has been defined to be dependent on the completion of “Clean F-20 board”. This last activity is not displayed when defining dependencies for “Clean F-20 board” to avoid circular dependencies and deadlocks.

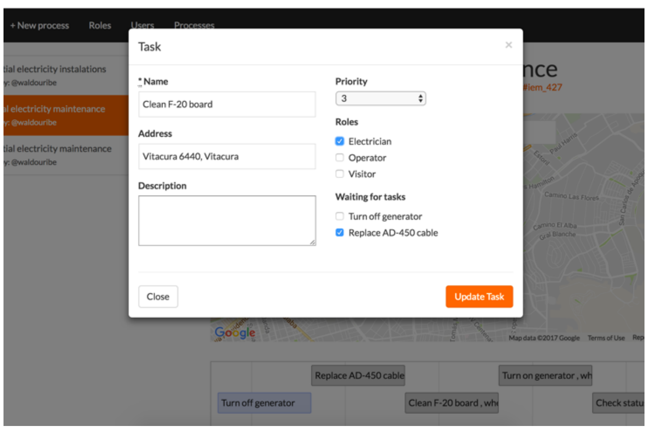


Figure 3: Defining attributes for an activity, including dependencies between activities.

4.2 Process execution

After describing the process enactment, we now describe the process execution. Users can pick activities from the worklist handler shown in Fig. 4. Users can select a process from the list shown at the top left part of the user interface, after which a list of available activities matching the user’s roles are displayed in a list with the header “Work to do”. Below this list, a map shows the locations where the activities should be performed. By clicking on an activity, the attributes are displayed in a pop-up window, similar to the one shown in Fig. 3. The user can then commit to execute the activity and the system will register the starting time. The activity will be immediately removed from the other users’ worklist handlers.

The order activities appear on the “Work to do” can be determined by two different criteria: priority number, and distance between user and activities.

Although the user interface shows all activities that can be executed by a user, not all of them might actually be selectable. Those activities which are dependent on other activities that have not been completed are shown with a darker color and are not selectable.

4.3 Twitting and process context

In Section 3, when discussing the process context, we noted that the Twitter platform could not only be used to support communication between the process participants but also as a way to attach contextual information to a process. In other words, instead of having a process model, we have a collection of tweets explaining how the process execution evolves over time. The tweets are automatically sent by the system to the Twitter platform. For this purpose, every time a user notifies the system that an activity has started or ended, the system contacts the REST API of Twitter and posts a tweet on the worker’s account. The structure of these messages is shown in Fig. 5.

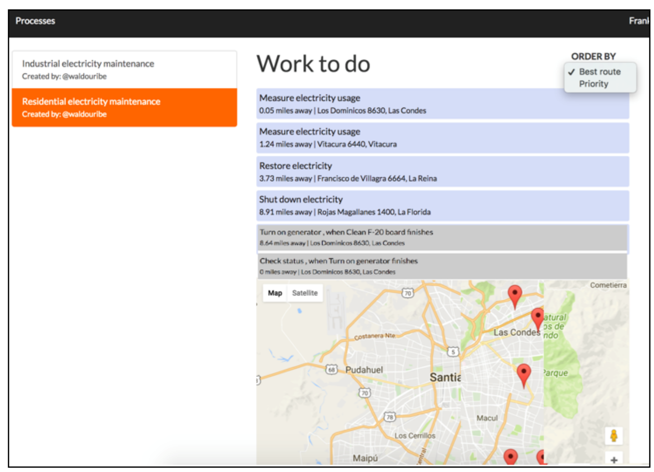


Figure 4: Worklist handler. The “Work to do” only shows the activities matching the user’s skills. The two activities shown in darker color have to wait the completion of other activities.

Fig. 6 is an actual screenshot from Twitter, which shows how messages are sent. We can see that user *frankjenson* has tweeted six messages while dealing with the *#item_407* process. (Note that date and time are implicit in the tweets.)

Of course, besides the semi-structured messages related to the #item_407 process, we should also expect to see unstructured messages needed to communicate about the process execution, like “I will start activity X late because of a traffic jam”, as explained in Section 3.

Starting activity:

@<username> twits
#<process id> <name of activity> started at
<address of activity> <date and time>

Ending activity:

@<username> twits
#<process id> <name of activity> ended <date and time>

Figure 5: Semi-structured messages sent to Twitter on behalf of the user.

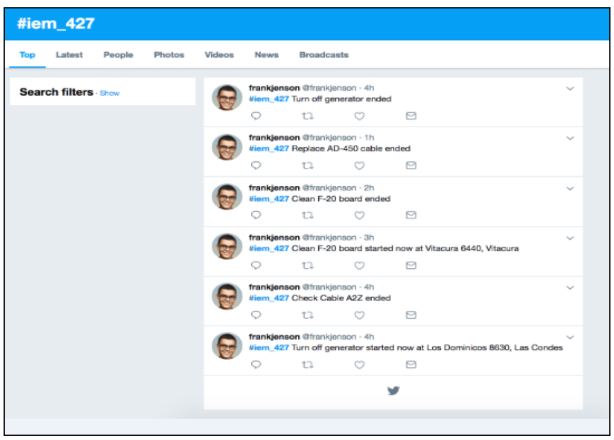


Figure 6: Twitter page of user frankjayson with tweets originated from the system communicating the starting and ending of activities for process item_427.

4.4 Twitting and process control

We specify process control in the following way:

@<username> **with role** <role list> **must** <activity> **at** <address> **when** <activity list> **finishes**

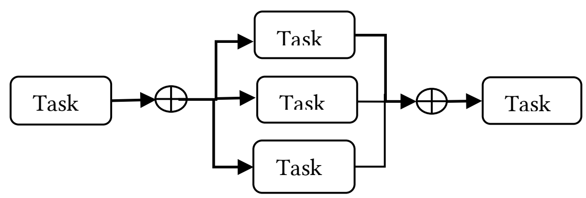
The character @ marks the beginning of an activity description, followed by the username that controls the activity execution. If no user is assigned, a question mark (?) is used instead. The keyword "with role" marks the beginning of the roles list which can perform the activity, separated by commas. The "at" keyword marks the beginning of the string containing the location where the activity has to be executed. The <activity list>, which is delimited by the keywords "when" and "finishes", contains the dependencies of this activity, i.e. other activities which have to be finished before this one can start. Except for the @ keyword, all other keywords in an activity description are optional.

It should be noted that the process name does not appear in these messages because the Twitter messages are stored as part of the process context. In Twitter, the messages have an associated hashtag which corresponds to the process name.

Fig. 7 illustrates process control using an example. The example consists of 5 activities with names Task1 to Task5 in which Task1 has to be completed before Task2, Task3 and Task4 can start, and Task5 has to wait for Task2, Task3 and Task4 before starting. The upper half of the figure shows the classical model-based control-flow specification, while the lower half shows the five sentences required to describe the same type of control using tweets. A fundamental difference between the two approaches is that the former is usually specified at design time, while the later can be dynamically defined as the process unfolds, and can be dynamically changed by sending tweets referring to the same activities but with different attributes.

The tweets are sent when an activity is created or edited (for changing attributes). They are automatically posted on Twitter using a hashtag with the process name.

Combining these tweets with the tweets notifying when an activity starts and ends (Fig. 8), allows to reconstruct the process enactment and execution, thus sharing the process context between the process participants. Currently, we have not yet implemented functionality to reenact a previously enacted process.



@user1 **must** Task 1 **at** address1
@user2 **must** Task 2 **at** address2 **when** Task1 **finishes**
@user3 **must** Task 3 **at** address3 **when** Task1 **finishes**
@? **must** Task 4 **at** address4 **when** Task1 **finishes**
@? **must** Task 5 **at** address5 **when** Task3, Task4, Task5 **finishes**

Figure 7: Above: model-based control-flow of 5 activities in which there are two "and" gates, one for diverging and another for converging the flow. Below: the same control using Twitter messages.

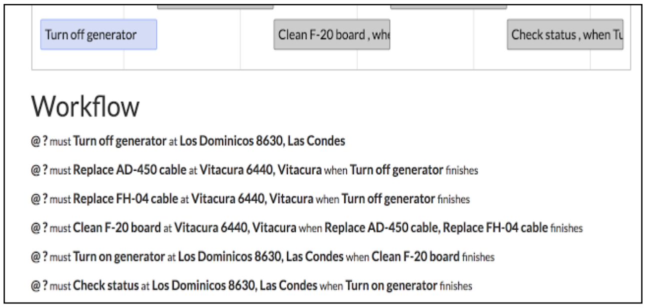


Figure 8: The tweets appear while editing an activity. They are automatically sent to Twitter under the hashtag representing the process.

5 PRELIMINARY, FORMATIVE EVALUATION

At this development stage, the main aspect we wanted to understand was the utility of having a system which can support people-driven processes for which the geographical location of activities is an important component. On a second level of importance, we also wanted to gather feedback on how to improve usability, which will be fundamental for conducting more formal evaluations in the future. Considering these requirements, a formative evaluation based on a focus group was selected.

The participants in the evaluation were recruited among computer engineering students who had already passed a course where they studied business process modelling, so they had background knowledge about the concepts of process structure and control. Five students were selected for the evaluation, 3 male and 2 female, all between 21 and 24 years old.

The evaluation was organized as follows. First, the system was described to the participants in a session of 30 minutes, and then they were asked to 1) think about a scenario in which the system would be helpful and 2) model in detail a concrete work process for such scenario. For this purpose, they had two days' time, after which the focus group session was organized to discuss the system support to the various scenarios and work processes elaborated by the participants.

The scenarios chosen by the participants included building a personal computer using parts from different stores, servicing a truck fleet, doing car repairs on the road, transportation of luxury cars, and even an arcade game with activities taking place in different parts of the world. To spark the focus group session, we raised two initial questions: “Do you think it is a good idea to geo-localize activities in work processes?” and “Do you think the system can be applied to the scenarios you developed?”.

All participants agreed that geo-located processes were a good idea and could be applied to several scenarios. The variety of the scenarios suggested and discussed by the participants reinforced the positive feedback.

To further discuss the system features, we also asked: “What do you think about adding priorities to activities?”, and “What do you think about adding roles to activities and filtering users according to them?”. Regarding priorities, the participants agreed that their value was not so clear, except for a specific scenario in which the users that create activities are not those that perform them. They regarded priorities as a kind of price that those who create activities are ready to pay to those performing them. Regarding roles, the participants agreed they could be of help in most scenarios.

To obtain feedback on the use of tweets to control processes, we also asked: “Do you think you could understand a process just by looking at the tweets generated by the system?”. Most participants agreed the tweets were understandable, but two of them noted that it takes some effort to reconstruct the whole process by only using the messages. They noted that a graphical view should be developed in the future.

To get feedback on how processes are enacted, we asked if process enactment was an easy task, to which most agreed it was not difficult at all. However, when asked if the user interface was intuitive enough, they said that it required some time to fully understand how to use it and that an initial explanation before using it was absolutely necessary.

When asked about which new features should the system provide, the participants suggested: to show on the map green pins for completed activities; to implement "drag-and-drop" to move activities on the map; to have a more accurate definition of activity locations; to be able to export a process as a BPMN model; and to check if users are near the activities' locations when reporting completion.

6 CONCLUSIONS

In this work, we present a system supporting people-driven, dynamic and geo-located work processes. The system allows the process participants to define activities in a dynamic way, while processes are being executed. To accomplish this, we substituted process models with process context; and substituted model-based control-flow with contextually-supported human control. We defined process context as a combination of activities and other attributes such as geographical location. This way, typical process models have been effectively substituted by process context. Process context ensures some desirable properties of BPM, such as visibility and traceability, while at the same time avoiding model-based control-flow, which constrains dynamic changes in process execution.

Our implementation uses the Twitter platform as a communication mechanism, allowing process participants to exchange semi-structured and unstructured messages about processes using familiar technology. Furthermore, we also use Twitter to support the process context: all messages exchanged about a process are stored in the platform and can be used both by the system and the process participants to understand how a process was enacted and how it evolved until completion.

This work makes the following unique contributions. Firstly, it provides an implementation of people-driven dynamic processes. Secondly, it provides an implementation combining people-driven dynamic processes with geographical locations. Thirdly, it provides a unique solution to building process context, leveraging the possibilities brought by microblogging platforms.

A preliminary, formative evaluation action conducted by a focus group suggested that the geographical contextualization of processes could be of great utility. Also, the obtained feedback supports using tweets for enacting and communicating about process, and assigning priorities to activities. However, the obtained feedback suggests that, even though it is not difficult to use the system, it requires some learning time to master it.

ACKNOWLEDGMENTS

This work was partially supported by Fondecyt Regular 1161200 and the 3rd Internal Research Contest FEN.

REFERENCES

[1] Barukh, M. and Benatallah, B. Processbase: A hybrid process management platform. In *International Conference on Service-Oriented Computing* (2014), 16-31.

[2] Antunes, P. and Mourão, H. Resilient business process management: Framework and services. *Expert Systems With Applications,* 38, 2 (2011), 1241-1254.

[3] Motahari-Nezhad, H. and Swenson, K. Adaptive case management: Overview and research challenges. In *IEEE 15th Conference on Business Informatics* (2013), 264-269.

[4] Gottanka, R. and Meyer, N. Modelasyougo:(re-) design of s-bpm process models during execution time. In *International Conference on Subject-Oriented Business Process Management* (2012), 91-105.

[5] Zhu, X., Recker, J., Zhu, G., and Santoro, F. Exploring location-dependency in process modeling. *Business Process Management Journal,* 20, 6 (2014), 794-815.

[6] Meyer, N. and Schiffner, S. Democratizing business process management. In *IEEE 16th Conference on Business Informatics* (2014), 93-100.

[7] Antunes, P., Zurita, G., and Baloian, N. Characterization of an ad-hoc geo-referenced bpm supported by microblogging. *IEEE Latin America Transactions,* 13, 6 (2015), 1989-1994.

[8] Kabicher-Fuchs, S., Rinderle-Ma, S., Recker, J., Indulska, M., Charoy, F., Christiaanse, R., and Mendling, J. Human-centric process-aware information systems (hc-pais). (2012).

[9] Schonenberg, H., Mans, R., Russell, N., Mulyar, N., and Van der Aalst, W. Process flexibility: A survey of contemporary approaches. In *Advances in enterprise engineering i* Heidelberg: Springer (2008), 16–30.

[10] Reichert, M. and Weber, B. *Enabling flexibility in process-aware information systems: Challenges, methods, technologies.* Heidelberg: Springer (2012).

[11] Kurz, M., Fleischmann, A., Lederer, M., and Huber, S. Planning for the unexpected: Exception handling and bpm. In *International Conference on Subject-Oriented Business Process Management* (2013), 123-149.

[12] Antunes, P. Bpm and exception handling: Focus on organizational resilience. *IEEE Transactions on System, Man, and Cybernetics Part C: Applications and Reviews,* 41, 3 (2011), 383-392.

[13] Pesic, M. Constraint-based workflow management systems : Shifting control to users Technische Universiteit Eindhoven. , Eindhoven (2008).

[14] Pesic, M. and Van der Aalst, W. A declarative approach for flexible business processes management. In *International conference on business process management* (2006), 169-180.

[15] Krumeich, J., Weis, B., Werth, D., and Loos, P. Event-driven business process management: Where are we now? A comprehensive synthesis and analysis of literature. *Business Process Management Journal,* 20, 4 (2014), 615-633.

[16] Scheer, A., Thomas, O., and Adam, O. Process modeling using event-driven process chains. In *Process-aware information systems* (2005), 119-146.

[17] Weidmann, M., Koetter, F., Kintz, M., Schleicher, D., Mietzner, R., and Leymann, F. Adaptive business process modeling in the internet of services (abis). In *Proceedings of the Sixth International Conference on Internet and Web Applications and Services* (2011).

[18] Brambilla, M., Fraternali, P., and Vaca, C. Bpmn and design patterns for engineering social bpm solutions. In *Bpm 2011 international workshops, clermont-ferrand, france, august 29, 2011*, Daniel, F., Barkaoui, K., and Dustdar, S., Eds. Heidelberg: Springer (2012), 219-230.

[19] Bider, I., Johannesson, P., and Perjons, E. A strategy for merging social software with business process support. In *Business Process Management Workshops* (2010), 372-383.

[20] Dorn, C., Dustdar, S., and Osterweil, L. Specifying flexible human behavior in interaction-intensive process environments. In *International conference on business process management*: Springer (2014), 366-373.

[21] Rothschädl, T. Ad-hoc adaption of subject-oriented business processes at runtime to support organizational learning. In *S-BPM ONE – Scientific Research* (2012), 22-32.

[22] Van der Aalst, W., Weske, M., and Grunbauer, D. Case handling: A new paradigm for business process support. *Data & Knowledge Engineering,* 53, 2 (2005), 129-162.

[23] Marin, M., Hauder, M., and Matthes, F. Case management: An evaluation of existing approaches for knowledge-intensive processes. In *International Conference on Business Process Management* (2015), 5-16.

[24] Herrmann, C. and Kurz, M. Adaptive case management: Supporting knowledge intensive processes with it systems. In *S-BPM ONE-Learning by Doing-Doing by Learning* (2011), 80-97.

[25] Böhringer, M. Emergent case management for ad-hoc processes: A solution based on microblogging and activity streams. In *Business process management workshops* Heidelberg: Springer (2011), 384-395.

[26] Dorn, C., Burkhart, T., Werth, D., and Dustdar, S. Self-adjusting recommendations for people-driven ad-hoc processes. *Business Process Management Journal,* (2010), 327-342.

[27] Dorn, C. and Dustdar, S. Supporting dynamic, people-driven processes through self-learning of message flows. In *Advanced information systems engineering* Berlin: Springer (2011), 657-671.

[28] Barukh, M. and Benatallah, B. Processbase: A hybrid process management platform. In *International conference on service-oriented computing* Heidelberg: Springer (2014), 16-31.

[29] Leoni, M., Adams, M., Van Der Aalst, W., and Hofstede, A. Visual support for work assignment in process-aware

information systems: Framework formalisation and implementation. *Decision Support Systems,* 54, 1 (2012), 345-361.

[30] Catarci, T., Leoni, M., Marrella, A., Mecella, M., Russo, A., Steinmann, R., and Bortenschlager, M. Workpad: Process management and geo-collaboration help disaster response. *International Journal of Information Systems for Crisis Response and Management,* 3, 1 (2011), 32-49.

[31] Zhu, X., Zhu, G., and Guan, P. Exploring location-aware process management. In *Geo-informatics in resource management and sustainable ecosystem* Heidelberg: Springer (2013), 249-256.

[32] Zhu, G. and Zhu, X. A study on geospatial constrained process modeling using uml activity diagrams. In *Asia-Pacific Conference on Business Process Management* (2014), 59-73.

[33] Benatallah, B., Sakr, S., Grigori, D., Motahari-Nezhad, H., Barukh, M., Gater, A., and Ryu, S. Business process paradigms. In *Process analytics*: Springer (2016), 19-60.

[34] Simões, D., Antunes, P., and Carriço, L. Eliciting and modelling business process stories: A case study. *Business & Information Systems Engineering,* (2017).

[35] Antunes, P., Simões, D., Carriço, L., and Pino, J. An end-user approach to business process modeling. *Journal of Network and Computer Applications,* 36, 6 (2013), 1466-1479.

[36] Simões, D., Antunes, P., and Cranefield, J. Enriching knowledge in business process modelling: A storytelling approach. In *Innovations in knowledge management: The impact of social media, semantic web and cloud computing.* vol. 95, Razmerita, L., Phillips-Wren, G., and Jain, L., Eds. Heidelberg: Springer (2016), 241-267.

[37] Van der Aalst, W. Business process management: A comprehensive survey. *ISRN Software Engineering,* (2013).

[38] Antunes, P., Zurita, G., Baloian, N., and Sapateiro, C. Integrating decision-making support in geocollaboration tools. *Group Decision and Negotiation*, 23, 2 (2014), 211-233.

[39] Monares, A., Ochoa, S.F., Pino, J.A., Herskovic, V., Rodriguez-Covili, J. and Neyem, A. Mobile computing in urban emergency situations: improving the support to firefighters in the field. *Expert Systems with Applications* 38(2) (2011), 1255-1267.

[40] Neyem, A., Franco, D., Ochoa, S.F. and Pino, J.A. Supporting Mobile Workflow with Active Entities. In *11th International Conference on Computer Supported Cooperative Work in Design* (2007), IEEE, 795-800.