Requirements for Ad-hoc Geo-referenced BPM with Microblogging

Pedro Antunes,[1] Gustavo Zurita,[2] Nelson Baloian[3]

[1] School of Information Management, Victoria University of Wellington
Wellington, New Zealand
pedro.antunes@vuw.ac.nz

[2] Department of Information Systems and Management, Universidad de Chile
Diagonal Paraguay 257, Santiago, Chile
gnzurita@facea.uchile.cl

[3] Department of Computer Science, Universidad de Chile
Av. Blanco Encalada 2120, Santiago, Chile
nbaloian@dcc.uchile.cl

Abstract. There are many scenarios in which business processes will benefit from the integration of geographical information for its management. In this paper we discuss a set of requirements for ad-hoc geo-referenced Business Process Management (BPM), noting in particular the conflicts between spatial and task dependencies when coordinating activities. We suggest the predominance of spatial dependencies and propose the integration of process models in georeferencing tools. We analyse the communication needs of geo-referenced and BPM processes and suggest the adoption of microblogging platforms for coordination support. We also discuss the implementation of an ad-hoc georeferenced BPM tool, specify the microblogging messages needed to coordinate georeferenced activities, and discuss a preliminary formative evaluation of the proposed implementation.

Keywords: BPM, GIS, geo-referenced processes, ad-hoc BPM.

1 Introduction

Many businesses require integrated process and geographical information management. For instance, in New Zealand it is common to contract private rubbish pickup companies, so that every week or on a designated date a truck comes by and picks up rubbish bins. Of course these contracts are constantly being created and cancelled, which means that collection routes have to be redefined on a weekly basis. Besides, the collection routes have to be permanently optimized. Geographical Information Systems (GIS) may help defining and optimizing routes in a visual way. But a route can also be seen as an ad-hoc business process [1] comprising a set of tasks. The term ad-hoc emphasizes the volatile nature of the business process, which requires constant redefinition.

A system integrating the functionality provided by GIS and Business Process Management (BPM) would provide support to people managing the scenario described above and many other similar scenarios such as infrastructure maintenance, processes for fire fighting, repair of street lighting, police rounds, and forest management, just to mention few. In this paper we discuss a set of requirements for ad-hoc geo-referenced BPM, noting in particular the conflict between spatial and task dependencies in coordination. We suggest the predominance of spatial dependencies and propose the integration of process models in geo-referenced tools. We analyse the communication needs of integrated GIS and ad-hoc BPM processes and also suggest the adoption of microblogging platforms for coordination support.

The paper is organized as follows. In Section 2 we discuss the requirements for geo-referenced BPM, focusing on the conflicts between spatial and task dependencies. In Section 3 we analyse the related research. Section 4 discusses the implementation of a geo-referenced BPM tool and specifies the set of microblogging messages necessary to coordinate geo-referenced activities. Finally, in Section 5 we present preliminary results from a formative evaluation action.

2. Requirements

BPM presupposes the existence of two core constituents: Process Aware Information Systems (PAIS) and process models [2]. The acronym PAIS does not refer to a particular system but to a category of systems that adopt a process view where business goals are decomposed in a discrete number of activities. This process view also introduces the notion of coordination through task-dependencies, i.e. activities are coordinated through precedence rules expressed by workflow patterns [3]. Process models specify the tasks and dependencies that are prototypical for a particular business process. They decouple process specification from enactment and allow implementing information systems based on high-level models that are easier to specify than low-level software code.

In turn, GIS presuppose two core constituents: data transformation and visualization [4]. Data transformation concerns the acquisition and representation of geographical data in the computer domain, while visualization supports spatial reasoning and decision-making. GIS induce spatial dependencies, i.e. work activities tend to be centred on geographical places or areas and coordination is implicitly related to changing the spatial focus of attention. Users do some reasoning activities related with a certain area and, when they "move" to another area, they implicitly start a different activity.

This dichotomy between task and spatial dependencies leads us to interrogate how work could be coordinated in an ad-hoc geo-referenced BPM system. Considering the rubbish collection example, a rubbish collection process can be modelled either by selecting a set of places and associating tasks to those places, or it can be modelled with a set of consecutive tasks, each one performed at a different place. Both models lead to the same result, but this example hints that the first choice is probably the best one, as tasks seem to be more contingent to spatial relationships. Effectively, we have been experimenting geo-collaborative tools in various scenarios [5-7] and have been observing that in most scenarios coordination is centred on spaces and not tasks, i.e. task dependencies are secondary to spatial dependencies. This leads to our first requirement:

R1. In an ad-hoc geo-referenced BPM system, coordination should be primarily determined by spatial dependencies and only secondarily determined by task dependencies.

This provides some flexibility as at least two different options may be considered regarding the structure of spatial data: either 1) the spatial model defines a path between different places, and therefore there is a sequence of points to traverse; or 2) there is no such path and users may select which region may be more convenient to work on. In any case we can view a business process as a collection of places or regions, each one having associated activities. GIS usually do not impose many task-dependencies and therefore we suggest that ad-hoc geo-referenced BPM should also not impose constraints on how users interact with spatial elements. This reasoning leads to the following two requirements.

R2. An ad-hoc geo-referenced BPM system should be regarded as an ad-hoc process where users determine the order of activities and places/regions provide context.

R3. Each place/region should have an associated sub-process.

R3 accommodates the combination of spatial reasoning and ad-hoc workflow, and can actually be modelled with current process modelling languages. For instance, in BPMN (Business Process

Management Notation) it means having a parent ad-hoc process with several sub-processes, one for each place/region, and where the affiliated sub-processes enclose a set of activities and dependencies related to that place/region.

Having suggested that a geo-referenced process should be an ad-hoc process, we have not yet committed our judgment about the sub-processes. In our rubbish collection example, we could consider several options. One is not detailing the activities related to collecting a rubbish bag, while a more extreme approach would define a very detailed sequence of activities related with stopping the vehicle on the road, deciding about collecting a bag or bin, ringing the bell if necessary (e.g. people with special needs), using the truck lift, and moving the vehicle. This later example is certainly an exaggeration but is presented here for illustrative purposes.

The research literature identifies categories of processes that allow us discussing the issue in more detail. These include tightly, loosely, ad-hoc and unframed processes [1]. Tightly framed processes have a model specifying all details about what and how activities should be accomplished. Loosely framed processes have a model describing normal behaviour but accept deviations such as skipping and repeating activities. Ad-hoc framed processes are unique, in the sense that the model may be constantly redefined. And finally, unframed processes do not specify any model and rely on collaboration to carry out the process.

These categories present very different requirements for geo-referenced BPM. For instance, tightly framed processes require mechanisms to reuse small process pieces, named worklets [8]. Loosely framed processes do not enforce control flow, i.e. all activities are available in the users' pools [9]. Ad-hoc framed processes require determining at each step what to do next. And unframed processes require informal communication support as a minimum basis for collaboration. We suggest the following requirements to address these issues.

R4. It should be possible to associate an unframed sub-process to a place/region where informal messages would be exchanged between the participants in that sub-process.

R5 Certain types of framed sub-processes may evolve according to tasks and control flow specified in runtime.

R4 and R5 highlight different messaging needs. In certain cases semi-structured messages with control flow events and constraints have to be exchanged, while in other cases unstructured messages have to be supported. Some messages involve communication between users, while others involve communication between users and system. We assert that the required types of communication can be supported through a microblogging platform, which main characteristics concern the capacity to send and receive short messages to and from a variety of destinations using a simple addressing mechanism [10]. In our case, a critical requirement is related with geographical referencing. This is expressed in our last requirement.

R6. An ad-hoc geo-referenced BPM system can be implemented on microblogging platform provided the exchanged messages are geographically referenced.

In Section 4 we discuss an implementation based on these requirements.

3. Related Work

3.1 Social BPM

The intersection between BPM and microblogging started receiving attention very recently and has been coined Social BPM [11,12]. An example is Tweetflows [13], a lightweight platform supporting the coordination of business processes using Twitter. The platform is aimed at crowdsourcing tasks and services using an ad-hoc approach where there is no process model and control flows are determined at runtime every time an activity finishes. The authors identify a set of primitives that support activity initiation and termination (called service request and response), besides other interesting primitives like delegating, which provide unusual flexibility to process enactment. The platform uses the typical "@" symbol to identify message receivers and hashtags to identify services. One problem pointed out by the authors is a lack of privacy/security, since messages are visible by all followers. More recently, the platform has been extended to support mobile workflow [14]. It does not support geo-referenced activities.

Böhringer [15] also addressed BPM support through microblogging, focusing again on ad-hoc processes and suggesting a close relationship between this type of process and several characteristics of social platforms such as a high degree of freedom and a more proactive approach to activity selection and execution. The author argues that by definition ad-hoc processes should not be modelled since modelling one single work instance creates unnecessary burden without benefits. The author presented the general concept of a prototype using hashtags to reference complete processes and the "@" symbol to represent human and automated activities. Following the principles suggested by case management, the proposed system does not include control flow. Instead users coordinate themselves by exchanging messages about a set of activities. Using hashtags it is possible to retrieve all message exchanged about a process.

Adaptive Case Management (ACM) has been suggested as the codename for the research area specifically concerned with the adoption of ad-hoc processes [16]. An example of an ACM system integrating Social BPM is Casebook [17]. Unlike the two systems mentioned above, Casebook provides a more heavyweight approach, structuring ad-hoc activities around cases and providing specific tools for case planning, case measuring, learning, and catalogue management.

3.2 GIS and BPM

Excluding cases where workflow techniques have been used to coordinate the computation of geographical information (e.g. [18]), which are out of our scope, the research literature on the integration of BPM and GIS is very scarce. Kaster et al. [19] and Weske et al. [20] developed a GIS with integrated decision support adopting a process view, but again this approach is out of scope since it does not address business processes in general and R1-3 in particular. Walter [21] suggests some potential advantages of using both types of systems for decision-making, for instance in the area of incident management. Incident management is an application area where the combination of framed and unframed processes could be beneficial, as it often requires combining planning and improvisation. Though we did not find examples in the literature explicitly implementing R1-3.

4. Implementation

4.1 Control flow

Van der Aalst et al. [3] suggested 20 patterns covering most BPM control flow needs. Some of these patterns are very complex and have not yet been adequately supported by current BPM systems, while others seem to be consensually implemented by most BPM systems. Considering the exploratory nature of this work, we opted to work with a minimal set of consensual patterns: Pattern 1 (Sequence), Pattern 2 (Parallel split), Pattern 3 (Synchronization), and Pattern 11 (Implicit termination).

These patterns can be implemented differently depending on the type of process framing considered (cf. Section 2). Consider for example that user U1 has completed task T1 and a sequence control flow is to be followed with T2 done by U2. Several possibilities can be discussed:

1. U1 notifies the workflow engine that T1 was completed. A worklet determines that T2 should be done by U2.
2. U1 notifies the workflow engine that T1 was completed. The workflow engine enables T2, which can be executed by U2 or any other allowed users.
3. U1 notifies a privileged agent (moderator) that T1 was completed. The agent determines that T2 should be done by U2.
4. U1 decides that T2 should be done by U2 and notifies U2.
5. U1 decides that T2 should be done next and notifies users that T2 is enabled, which U2 may offer to execute.
6. U1 notifies users that T1 was completed. Users may discuss the issue to determine that T2 should be done next by U2, or maybe U2 offers to execute T2.

Option 1 reflects the typical behaviour of a tightly framed process. Option 2 implements a loosely framed process. Options 3-5 reflect different alternatives to implement the typical behaviour of ad-hoc framed processes. And option 6 is associated with unframed processes.

Analysing the above options in more detail, we decided not to implement option 3, since centralizing control flow decisions would require too much burden from the privileged agent. Option 5 has also been discarded as too similar to the unframed strategy.

The possibilities discussed above, which consider a sequence pattern, could also be extended to the other patterns without many surprises. The exception concerns synchronization in the context of ad-hoc framed processes. The question is who synchronizes tasks in an ad-hoc context. Let us consider an example where T1 done by U1 and T2 done by U2 should be synchronized before starting T3 done by U3. Several possibilities can be considered for control flow in an ad-hoc framed process:

1. U1 and U2 notify a privileged agent that the respective tasks were completed. The agent determines that T3 should be done by U3.
2. U1 decides that T2 should be merged and notifies U2. After receiving a notification from T2, U1 decides that T3 should be done by U3 and notifies U3. Symmetrically the same decision can be made by U2.
3. The last alternative is slightly more complex as it requires considering the parallel split that originated the two parallel flows that will be synchronized. We can consider that the system may require the user specifying a parallel split should also specify the corresponding synchronization.

Control flow	Ad-hoc framed process	Unframed process
Sequence	U → U	G → U
Parallel split	U → U+	G → U+
Synchronization	U → U+ - U	U+ → G

Table 1. Adopted control flow mechanisms. U means “user”; U+ means “one or more users”; G means “group”; and → means “transition”. E.g. U → U means “passing control from one user to another”.

Analysing these three alternatives, we note that having a privileged agent making control flow decisions is against the “spirit” of a pure ad-hoc approach. We can make the same comment about the last alternative, which requires users to pre-determine synchronizations when defining splits. Such an approach violates the principle that an ad-hoc framed process can be redefined as the process enfolds. We therefore decided to only implement the option 2 mentioned above. Table 1 summarizes the adopted control flow mechanisms.

4.2 Messaging for ad-hoc and unframed process

As previously discussed, informal communication and control flow require informal and semi-formal messaging, respectively. We adopted Twitter to implement messaging. We use the typical "#hashtag" to refer to specific business process instances, and the also typical "@user" to refer to participants, including the workflow engine. The "%number" tag was adopted to refer to van der Aalst's workflow patterns, e.g. %1 refers to sequence and %2 to parallel split. The message specification is provided in Table 2 using regular expressions. Note that geographical locations are specified with Twitter's location application interface and therefore do not appear in the message.

twitter message = code? comment code = sequence | and-split | and-join-start | and-join-end | termination sequence = %1 process agent? and-split = %2 process flow agent and-join-start = %3a process flow and-join-end = %3b process flow agent	termination = %11 process flow? process = #\w flow = \^\w agent = @\w comment = .*

Table 2. Twitter messages necessary to implement the ad-hoc geo-referenced BPM tool.

One interesting characteristic of the adopted messaging scheme is that we do not explicitly refer to tasks but to users. A task is implicitly defined by sending a message to a user. The second characteristic is that parallel flows initiated by splits are explicitly named using the "^flow" tag. Let us illustrate how the messaging scheme works with the fire-fighting example shown in Table 4.

	Sent from	Message	Locations
1	FFOfficeC1	%1 #Ruta68 @FFC1Captain Auto driver reports fire on route 68, 10 km east from Valparaiso	L1(msge)
2	FFC1Captain	#Ruta68 On our way	L1(msge)
3	FFC1Captain	#Ruta68 Arrived to fire site	L1(msge)
4	FFC1Captain	%2 #Ruta68 ^FireCloseRoad @Firefighter1C1 Attack fire close to the road	L1.1(^FireCloseRoad)
5	FFC1Captain	%2 #Ruta68 ^BuildFirewall @Firefighter2C1 Build firewall between road and fire	L1.2(^BuildFirewall)
6	FFOfficeC2	%1 #Ruta68 @FFC2Captain House owner reports trees are catching fire near house before arriving to Valparaiso near route 68	L2(msge)
7	FFC2Captain	#Ruta68 We are on our way	L2(msge)
8	FFC2Captain	#Ruta68 Arrived to fire site	L2(msge)
8	FFC2Captain	%2 #Ruta68 ^FireOnTrees @Firefighter1C2 Extinguish fire from trees	L2.1(^FireOnTrees)
9	FFC2Captain	%2 #Ruta68 ^BuildFirewall2 @Firefighter2C2 Build firewall between trees and house	L1.2(^BuildFirewall2)
10	Firefighter1C1	%1 #Ruta68 @FFC1Captain ^FireCloseRoad Fire extinguished	L1.1(^FireCloseRoad)
11	FFC1Captain	%3a #Ruta68 ^FireCloseRoad Stand by	L1.1(^FireCloseRoad)
12	Firefighter1C1	%3b #Ruta68 ^FireCloseRoad @FFC1Captain OK	L1.1(^FireCloseRoad)
13	Firefighter2C1	%1 #Ruta68 @FFC1Captain ^BuildFirewall Captain, firewall done	L1.2(^BuildFirewall)
14	FFC1Captain	%3a #Ruta68 ^BuildFirewall Stand by	L1.2(^BuildFirewall)
15	Firefighter2C1	%3b #Ruta68 ^BuildFirewall @FFC1Captain OK	L1.2(^BuildFirewall)
16	Firefighter1C2	%1 #Ruta68 @FFC2Captain Need help fighting fires on trees	L3(msge)
17	FFC2Captain	%1 #Ruta68 @FFC1Captain Need help fighting fires on trees	L3(msge)
18	FFC1Captain	%1 #Ruta68 @Firefighter1C1 Help fire on trees	L3(msge)
19	FFC1Captain	%1 #Ruta68 @Firefighter2C1 Help fire on trees	L3(msge)
20	Firefighter1C2	%1 #Ruta68 @FFC2Captain Fire on trees extinguished	L3(msge)
21	Firefighter2C2	%1 #Ruta68 @FFC2Captain Firewall done	L1.2(^BuildFirewall2)
22	FFC2Captain	#Ruta68 Going to check people in house	L2(msge)

23	FFC2Captain	#Ruta68 Everyone ok, returning to headquarters	L2(msge)
24	FFC1Captain	#Ruta68 Ok, returning to headquarters	L1(msge)
25	FFOfficeC1	%11 #Ruta68 Action completed	L1(msge)

Table 3. Exchanged messages for a fire fighting ad-hoc process. Ln means a "n" geo-referenced location, Lnm is a "m" geo-referenced location/region close to "n". Ln(^flow) is a "n" geo-referenced location of a flow "flow". Ln(msge) is a geo-referenced location associated to a message.

In this example, fire appears near the road and two companies are called to extinguish the fire. Tasks are assigned to fire fighters as described in Table 3. In this case, locations are associated to flows or places where sub-processes are performed (see column "Locations" in Table 3). They represent the spatial dependencies of tasks. Messages 1-3, 6-8, 16-20 and 22-24 require the association of the location information. For example, in message 1, it is necessary to inform where the fire is taking place; in message 6, the exact place where a fire is notified should be indicated. The messages that have no associated task or that do not require changing locations are associated to the locations mentioned in the previous messages (e.g., messages 2 and 3 take place in the same location as message 1; messages 7 and 8 take place in the same location as message 6). If a change of location is needed, the new message has to explicitly reference the new location context (e.g., message 6 changes the location context to L2). When tasks need to be geo-referenced, they should be associated to a location in the map. In messages 4 and 5 of our example, tasks ^FireCloseRoad and ^BuildFirewall take place in locations L1.1 and L1.2, respectively, near location L1 were fire was reported for the first time. In this way fire fighters know exactly where they have to move to perform the task.

5. Preliminary evaluation and discussion

A scenario based evaluation technique was adopted to assess the viability of the fire-fighting example described in Section 4. Running the scenario with several users allowed understanding the system's perceived usability and utility, the communication and coordination problems users would have using the system, and the design options that should be further explored at the next development stages. We evaluated the fire-fighting scenario with four users. The users walked through the messages detailed in Table 3 considering a realistic fire-fighting setting. We then conducted post-session interviews with the users, focussing especially on communication and coordination breakdowns.

The formative evaluation allowed a better understanding of various design issues: a) It is better using a minimal set of workflow patterns, to keep the system simple and understandable by people unfamiliar with BPM. In particular, the users referred that the approach was simple enough, while having a high potential for describing ad-hoc activities; b) The association of messages to locations may not be required when tasks take place in the same location, but are very useful when tasks are distributed; c) There are multiple ways to specify the coordination of the fire-fighting process. Some specifications use simple, informal messages, like in a normal conversation. Though the users agreed that often some explicit coordination is needed; d) The messaging mechanism affords users with the desired level of detail necessary to accomplish different tasks. Though users suggested that each message could be expanded to allow a deeper level of detail, especially regarding the task description.

References

1. van der Aalst W (2013) Business Process Management: A Comprehensive Survey. ISRN Software Engineering

2. Weber B, Reichert M, Rinderle S (2008) Change patterns and change support features – Enhancing flexibility in process-aware information systems. Data & Knowledge Engineering 66 (3):438-466
3. Van der Aalst W, Hofstede A, Kiepuszewski B (2003) Workflow Patterns. Distributed and Parallel Databases 14:5-51
4. Goodchild M (2010) Twenty years of progress: GIScience in 2010. Journal of Spatial Information Science 1:3-20
5. Antunes P, Sapateiro C, Zurita G, Baloian N (2010) Integrating Spatial Data and Decision Models in a E-Planning Tool. In: Groupware: Design, Implementation, and Use. 16th CRIWG Conference on Collaboration and Technology, Maastricht, The Netherlands, vol 6257. Springer, Heidelberg, pp 97-112
6. Antunes P, Zurita G, Baloian N, Sapateiro C (2014) Integrating Decision-Making Support in Geocollaboration Tools. Group Decision and Negotiation 23 (2):211-233
7. Antunes P, Zurita G, Baloian N (2014) Key Indicators for Assessing the Design of Geocollaborative Applications. International Journal of Information Technology & Decision Making 13 (2):361-385
8. Adams M, Hofstede A, Edmond D, Van der Aalst W (2006) Worklets: a service-oriented implementation of dynamic flexibility in workflows. In: Meersman R, Zahir T (eds) On the Move to Meaningful Internet Systems 2006: CoopIS, DOA, GADA, and ODBASE, OTM Confederated International Conferences, CoopIS, DOA, GADA, and ODBASE 2006, vol 4275. Springer, Heidelber, pp 291-308
9. van der Aalst W, Weske M, Grunbauer D (2005) Case handling: A new paradigm for business process support. Data & Knowledge Engineering 53 (2):129-162
10. Honey C, Herring S Beyond microblogging: Conversation and collaboration via Twitter. In: 42nd Hawaii International Conference on System Sciences, Hawaii, 2009. IEEE, pp 1-10
11. Brambilla M, Fraternali P, Vaca C (2012) BPMN and design patterns for engineering social BPM solutions. In: Business Process Management Workshops. Springer, Heidelberg, pp 219-230
12. Erol S, Granitzer M, Happ S, Jantunen S, Jennings B, Johannesson P, Schmidt R (2010) Combining BPM and social software: contradiction or chance? Journal of software maintenance and evolution: research and practice 22 (6-7):449-476
13. Treiber M, Schall D, Dustdar S, Scherling C Tweetflows: flexible workflows with twitter. In: Proceedings of the 3rd International Workshop on Principles of Engineering Service-Oriented Systems, 2011. ACM, pp 1-7
14. Treiber M, Schall D, Dustdar S, Scherling C Creating mobile ad hoc workflows with Twitter. In: Proceedings of the 27th Annual ACM Symposium on Applied Computing, 2012. ACM, pp 1998-2000
15. Böhringer M (2011) Emergent case management for ad-hoc processes: a solution based on microblogging and activity streams. In: Business Process Management Workshops. Springer, Heidelberg, pp 384-395
16. Motahari-Nezhad H, Swenson K Adaptive Case Management: Overview and Research Challenges. In: IEEE 15th Conference on Business Informatics (CBI), 2013. IEEE, pp 264-269
17. Motahari-Nezhad H, Spence S, Bartolini C, Graupner S, Bess C, Hickey M, Rahmouni M (2013) Casebook: A Cloud-Based System of Engagement for Case Management. IEEE Internet Computing 17 (5)
18. Chen Q, Wang L, Shang Z MRGIS: A MapReduce-Enabled high performance workflow system for GIS. In: IEEE Fourth International Conference on eScience, 2008. IEEE, pp 646-651
19. Kaster D, Medeiros C, Rocha H (2005) Supporting modeling and problem solving from precedent experiences: the role of workflows and case-based reasoning. Environmental Modelling & Software 20 (6):689-704
20. Weske M, Vossen G, Medeiros C, Pires F Workflow management in geoprocessing applications. In: Proceedings of the 6th ACM international symposium on Advances in geographic information systems, 1998. ACM, pp 88-93
21. Walter M Situational Awareness for Enhanced Incident Management (SAFE-IM). In: Military Communications Conference, 2007. IEEE, pp 1-6