

CAPE : Context-Aware Agile Business Process Engine

Irina Rychkova, Manuele Kirsch-Pinheiro, Bénédicte Le Grand
Centre de Recherche en Informatique, Université Paris 1, Panthéon-Sorbonne

Résumé de la présentation orale

L'avenir de la gestion des processus d'entreprise va voir évoluer des systèmes actuels reposant sur les *workflows* vers des structures dynamiques agiles sensibles au contexte, qui exploitent l'adaptabilité locale. Afin de permettre cette agilité, nous proposons notre vision de *Business Process Management* sensible au contexte, reposant sur la modélisation déclarative combinée à une gestion de contexte innovante et à l'Analyse Formelle de Concepts. Au cours de cette présentation, nous décrirons les fondements et l'architecture d'un moteur de processus métier agile et sensible au contexte.

Résumé étendu

La capacité à découvrir rapidement et à répondre efficacement aux changements de l'environnement est l'objectif majeur de l'entreprise du futur. Selon [10][4], la faculté d'une entreprise à s'adapter aux environnements dynamiques dépend tout d'abord de l'agilité de ses processus métier. Il est donc essentiel de pouvoir concevoir et développer de nouveaux systèmes de gestion de processus permettant l'adaptation des processus en cours d'exécution.

Lamport[7] définit un processus comme une séquence d'événements qui se produisent dans le système, où chaque événement est déclenché par une action. Un processus métier peut ainsi être vu comme une suite d'événements déclenchés par les activités des acteurs. La majorité des méthodes existantes de conception de processus métier suivent des principes impératifs, ce qui implique la prédétermination de l'ordre des événements. Par conséquent, tous les événements significatifs doivent être prédéterminés au moment de la conception, de même que les actions correspondantes. Au moment de l'exécution, les processus suivent alors le modèle configuré avec des possibilités limitées de dévier du scénario prédéfini.

Nous définissons la **première forme d'agilité d'un processus métier** comme la capacité à tenir compte d'événements imprévus dans le système. Ceci implique que l'ordre d'invocation des activités soit défini de manière dynamique, lors de l'exécution, et que cet ordre ne dépende que de la situation courante (état des processus), plutôt que d'un scénario prédéfini. Afin de permettre cette première forme d'agilité, nous changeons le paradigme impératif traditionnel de la conception de processus et nous exploitons des principes déclaratifs : nous représentons un processus métier sous la forme d'un automate à états finis [9], où un état représente la situation d'un processus à un instant donné et où les transitions entre les états définissent les scénarios possibles. Les événements déclencheurs spécifient la sémantique sous-jacente, c'est-à-dire les conditions nécessaires aux transitions d'états. Le formalisme des automates à états finis rend implicite la notion d'activité de processus, tout en mettant en avant les résultats de l'activité, qui sont modélisés comme des événements déclencheurs. Le modèle de processus déclaratif se concentre ainsi sur « ce qui a besoin d'être fait » pour atteindre l'objectif du processus, et non sur « la manière dont cela doit être fait ».

Ceci nous permet de gérer des événements dont l'ordre d'apparition n'est pas déterminé à l'avance, et de définir à l'exécution les scénarios correspondants.

Dans notre modèle, l'ordre partiel entre les activités des processus est déterminé par la relation de transition d'états. Tout comme [1], nous utilisons le terme de « navigation » pour décrire la manière dont un processus devrait s'exécuter. Nous suggérons qu'au lieu de suivre un scénario d'exécution prédéterminé, un processus navigue dans un « espace d'états » et ajuste dynamiquement sa trajectoire en fonction de l'état courant, de la situation courante et des règles de navigation. Nous proposons de concevoir des règles de navigation pour guider les processus grâce à l'Analyse Formelle de Concepts et aux treillis de Galois [2][5]. Les treillis de Galois permettent en effet de mettre en correspondance les événements déclencheurs de transition d'état et les activités pouvant conduire à l'apparition de ces événements. Les règles de navigation permettent de recommander un processus pour atteindre l'état visé ; nous spécifions le processus résultant de cette recommandation comme un ensemble d'activités qui peuvent être assemblées dynamiquement lors de l'exécution dans un scénario. Une telle spécification peut offrir un grand nombre de scénarios alternatifs et la possibilité de passer d'un scénario à un autre au cours de l'exécution. Dans l'approche que nous proposons, les états des processus, les événements déclencheurs et les activités des processus constituent un *contexte formel* et peuvent être analysés grâce à un treillis de Galois construit à partir de ce contexte. Les états et les activités peuvent être regroupés en des clusters qui révèlent leurs propriétés conceptuelles. Par exemple, on peut déterminer (ou suggérer) les activités qui peuvent être exécutées sous certaines conditions et dans le but de déclencher une transition d'états particulière. Les treillis de Galois constituent un outil de raisonnement compatible avec l'approche déclarative, avec des descriptions sémantiques telles que des ontologies, et avec des modèles de contexte nécessaires pour la 2^e forme d'agilité décrite ci-dessous.

Nous définissons la **deuxième forme d'agilité des processus métier** comme la capacité de contrôler et de gérer le contexte des processus et de modifier le scénario d'exécution en fonction de ce contexte. Nous étendons les spécifications déclaratives des processus grâce à des modèles et des mécanismes de gestion de contexte dynamiques [3][6]. Les paramètres du contexte reflètent la prise en compte des informations (internes et externes) relatives aux processus ; ces paramètres peuvent être observés et mesurés. Bien que la sensibilité au contexte des processus métier soit déjà étudiée dans la littérature [11][8], il manque toujours un formalisme pour la représentation et la gestion du contexte : la plupart des modèles proposés sont statiques (i.e. définis dès la conception), incomplets (ils ne prennent en compte qu'une information de contexte limitée) et souvent spécifiques aux processus de *typeworkflow*.

Selon nous, le nombre et le type des paramètres du contexte peuvent varier d'une situation (ou d'un état de processus) à l'autre, ce qui rend impossible la modélisation exhaustive de l'information de contexte dans un unique modèle statique. Ainsi, le modèle de contexte doit lui-même être instancié dynamiquement à partir d'un meta-modèle, en fonction des dimensions de contexte spécifiques (et évolutives).

Les deux formes d'agilité présentées ici visent à relaxer les exigences à la fois :

- du point de vue de l'expertise humaine pour permettre une certaine flexibilité des processus, et

- du point de vue de la rigidité des processus (structure prédéfinie) pour fournir le niveau d'automatisation souhaité.

L'approche déclarative pour la conception de processus, couplée à des méthodes formelles, mène à un ensemble de techniques automatisées pour l'analyse et la validation des processus, reposant sur la vérification de modèle et la preuve de théorèmes. Cette approche améliore ainsi le niveau d'automatisation de l'aide à l'utilisateur, en fournissant la flexibilité maximale durant l'exécution. La sensibilité au contexte et l'Analyse Formelle de Concepts permettent d'automatiser les recommandations et l'identification de scénarios alternatifs. Leur utilisation conjointe fournit un guidage flexible à l'utilisateur final en cours d'exécution et leur fournit l'expertise requise pour gérer les processus.

Cette combinaison innovante de principes de modélisation déclarative, de modélisation dynamique du contexte et d'Analyse Formelle de Concepts constitue la principale contribution de ce travail. Nous avons en effet proposé un modèle et une méthode pour la spécification de processus métier agiles reposant sur l'abstraction des automates à états finis et l'Analyse Formelle de Concepts, et nous avons étendu ce modèle avec un modèle de contexte dynamique.

Notre approche est synthétisée dans la Figure 1, qui représente l'architecture d'un moteur de processus métier agile et sensible au contexte.

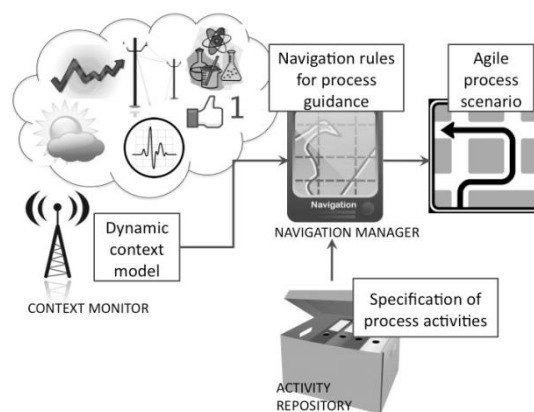


Fig. 1. CAPE architecture: the context monitor, the activity repository and the navigation manager.

Références

1. Andersson, T., Bider, I., Svensson, R.: Aligning people to business processes experience report. *Software Process: Improvement and Practice*, vol.10, no.4 pp. 403-413 (2005)
2. Barbut, M., Monjardet, B. : *Ordre et classification. Algèbre et combinatoire*, Tome 2, Hachette (1970)
3. Bettini, C., Brdiczka, O., Henricksen K., Indulska, J., Nicklas, D., Ranganathan, A., Riboni, D.: A survey of context modelling and reasoning techniques, *Pervasive and Mobile Computing*, vol. 6, issue 2, pp. 161-180 (2010)
4. Bider, I., Johannesson, P., Perjons, E.: Do workflow-based systems satisfy the demands of the agile enterprise of the future? In: Rosa, M.L., Soffer, P. (eds.), *Business Process Management Workshops (BPM 2012)*, LNBP, vol. 132, pp.59-64, Springer (2013)

5. Birkhoff, G.: Lattice Theory, First Edition, Amer. Math. Soc. Pub. 25, Providence, R. I. (1940)
6. Dey, A.: Understanding and Using Context, Personal and Ubiquitous Computing, vol. 5, pp. 4-7 (2001)
7. Lamport, L.: Time, clocks, and the ordering of events in a distributed system. CACM, vol. 21, no. 7 (1978), pp. 558-565
8. Mounira, Z., Mahmoud, B.: Context-aware process mining framework for Business Process flexibility, iiWAS2010, Paris (2010)
9. Plotkin, Gordon D.: A structural approach to operational semantics (1981).
10. Raschke, R. L., David, J.S.: Business process agility. (2005)
11. Rosemann, M., Recker, J. Flender, C.: Contextualization of Business Processes, Int. J. Business Process Integration and Management, vol.1, n°1, pp. 46-60 (2007)