Variability in Business Process Families

Colette Rolland, Université Paris 1 – Panthéon – Sorbonne, Centre de Recherche en Informatique, 90 rue de Tolbiac, 75634 Paris cedex 13, France; E-mail: rolland@univ-paris1.fr

Naveen Prakash, Université Paris 1 – Panthéon – Sorbonne, Centre de Recherche en Informatique, 90 rue de Tolbiac, 75634 Paris cedex 13, France; E-mail: praknav@hotmail.in

Rim-Samia Kaabi, Université Paris 1 – Panthéon – Sorbonne, Centre de Recherche en Informatique, 90 rue de Tolbiac, 75634 Paris cedex 13, France; E-mail: rkaabi@univ-paris1.fr

ABSTRACT
Variability proved to be a central concept in different domains, manufacturing, software development etc. in order to develop solutions that can be easily adapted to different organisational settings and different sets of customers at a low price. We argue that families of business process models can facilitate the installation of situated models in different organisations. We propose a representation system called MAP to capture variability in process models expressed in an intentional manner through business goals and strategies. The paper presents MAP and illustrates its use in an excerpt of a real case in the Electricity Supply Industry.

1. INTRODUCTION
Variability as a concept has proved to be useful in various engineering domains in which companies are not anymore faced to the development of a single product but to engineering product lines and families. The former represent the evolution of a given product, for example, a DVD player, whereas the latter integrates different product lines having commonality, such as DVD and MP3 player lines. Variability has been introduced to explicitly differentiate between the common and different parts in a set of similar but different product lines of a product family. Managing commonalities and variability leads to two major advantages:

a. reuse of common parts (Ommering, 2002; Tomphson, 2001) and,

b. adaptation of products to different customers and various organisational settings (Svahnberg, 2001).

Seeing the duality that exists between products and processes, our position is that business process families do exist today in companies and could beneficially be handled by introducing the concept of variability. Indeed, since the process wave initiated by Hammer and Champy (Hammer, 1994), large portfolios of business process models have been created and continue to develop as business process models are recognised as indispensable artefacts to driving business management and evolution. Besides, these portfolios evolve due to internal factors leading to business process evolution and/or external factors or mergers and acquisitions where different processes, perhaps having common parts, have to be integrated. Further, business process change is not just the replacement of one process by another. Rather process change management involves the reuse of parts of the process to be discarded, inclusion of parts of other processes, co-existence of different versions of the same process etc. As a matter of fact, one can recognize business process lines and families in current organizations today.

The foregoing suggests a move away from management of individual process to managing a set of similar processes considered as a whole, a family. Our proposal is to organize business processes as business process families and to manage variability and commonalities within the family in order to promote reuse and adaptability of business process models. We understand a business process family to be a collection of processes meeting a common goal but in different ways. For example, the goal ‘admit students’ can be achieved through a business process family comprising three processes that select students on the basis of a national entrance examination, a university test, or school performance respectively. The variability across the three processes is obvious. However, there is a commonality between these three processes as well: all these processes have to accept fees from the admitted student.

In this paper, we propose a modeling formalism called MAP to capture variability across business processes of a family in an intentional manner. The map is a directed, labeled, non-deterministic graph with goals as nodes, and strategies to achieve goals, as edges. Its nature allows the capture of different forms of variability through multi-edges between a pair of nodes thereby enabling many different traversals of the graph from beginning to end. Besides, using the refinement mechanism of the map, it is possible to represent variability at different levels of detail, in a hierarchy of maps. We show that this hierarchical nature permits us to represent process families as maps. We also show the power of a map to represent variability and, as an illustration, model the variations of an electricity supply process family as a hierarchy of maps.

The paper is organized in three sections. The next section introduces the MAP formalism and shows how it captures business feature variability. Section 3 presents an example and section 4 considers the adaptation of a business process model within a family.

2. CAPTURING BUSINESS VARIABILITY IN MAPS
We use the MAP formalism (Rolland, 2000) to capture variability of business processes modelled in an intentional manner.

2.1 Business Intentionality in Maps
A map is a process model expressed in a goal driven perspective. Map provides a process representation system based on a non-deterministic ordering of goals and strategies. A map is represented as a labeled directed graph (see an example in Fig. 6) with goals as nodes and strategies as edges between goals. The directed nature of the graph shows which goals can follow which one.

A goal can be achieved by the performance of a process. Each map has two special goals, Start and Stop to start and end the process respectively.

A strategy is an approach, a manner to achieve a goal. A strategy is a triplet $S_{ij}$ between the couple of goals $G_i$ and $G_j$ represents the way $G_j$ can be achieved once $G_i$ has been satisfied.

A section is a triplet $<G_i, G_j, S_{ij}>$ and represents a way to achieve the target goal $G_j$ from the source goal $G_i$ following the strategy $S_{ij}$. Each section of the map captures the situation needed to achieve a goal and a specific manner in which the process associated with the target goal can be performed.

A section in a map can be refined as a map (see Fig.7). This leads to intentional process modeling as a hierarchy of maps.

As process models, maps can be compared to the various types of process modelling languages and formalisms that have emerged supporting a variety of purposes. The existing formalisms can be roughly classified according to their orientation to activity-sequence oriented languages (e.g., UML Activity Diagram), agent-oriented languages (e.g., Role-Activity Diagram (Ould, 1995)), state-based languages (e.g. UML state charts), an intention-oriented languages (e.g. Maps). The concept of goal is central in business process modelling and design. It is included in many definitions of business processes (e.g. “a business process is a set of partially ordered activities aimed at reaching a goal” (Hammer, 1994)). However, most process modelling languages do not employ a goal construct as an integral part of the model. This is sometimes justified by viewing these models...
as an “internal” view of a process, focusing on how the process is performed and externalising what the process is intended to accomplish in the goal (Dietz, 2004).

In contrast, intention-oriented process modelling focuses on what the process is intended to achieve, thus providing the rationale of the process, i.e., why the process is performed. Intention-oriented process modelling such as MAP, follows the human intention of achieving a goal as a force, which drives the process. As a consequence, goals to be accomplished are explicitly represented in the process model together with the alternative ways for achieving them, thus facilitating the selection of the appropriate alternative for achieving the goal.

2.2 Modeling Business Variability in Maps

For the sake of conciseness, we use a textual notation in which goals are named by letters of the alphabet, strategies are numbers and therefore, a section named ab1 designates a way to achieve a target goal b from a source one a following a strategy i. Thus, the section <Gi, Gi, Sij> is named ab, where a is the code of the goal Gi, b is the code of the goal Gj, and i is the code of the strategy Sij (see Fig. 1).

We advocate that sections are at the right abstraction level to capture business variability. We consider a section as an important process characteristic that business agents (managers, decision makers, actors...) want the business to provide and also an abstraction of a business flow. By analogy with software variability, a section can be related to the notion of a feature In FODA (Kang, 1990) for example, a feature is defined as “A prominent or distinctive user-visible aspect, quality or characteristic of a software system or systems”. In (Bosch, 2001), a feature is “a logical unit of behavior that is specified by a set of functional and quality requirements”. The point of view taken in this paper is that a business feature is a representation of a visible process characteristic and an abstraction of a cohesive business flow of activities expressed in an intentional manner.

Features represented in a map are related to each others by four kinds of relationships namely multi-thread, bundle, path and multi-path relationships. The relationships show the possible combination of features from which a business agent can select the appropriate ones according to the situation at hand. Let us now see how these relationships are used to express variability in business models.

The multi-thread relationship: when there are various ways to achieve the same goal starting from a source, features are related by a multi-thread relationship.

A multi-thread relationship is represented in a map by several strategies between a goal and also an abstraction of a business flow. By analogy with software variability, a section can be related to the notion of a feature In FODA (Kang, 1990) for example, a feature is defined as “A prominent or distinctive user-visible aspect, quality or characteristic of a software system or systems”. In (Bosch, 2001), a feature is “a logical unit of behavior that is specified by a set of functional and quality requirements”. The point of view taken in this paper is that a business feature is a representation of a visible process characteristic and an abstraction of a cohesive business flow of activities expressed in an intentional manner.

Figure 1. A section

\[
\begin{align*}
S_{ij} & \quad \text{Gi} \quad \text{Gj} \\
S_{ij} & \quad \text{Gj} \\
Gj & \quad \text{a} \\
Gj & \quad \text{b}
\end{align*}
\]

Figure 2. A multi-thread relationship

\[
\begin{align*}
S_{ij1} & \quad \text{Gj} \\
S_{ij2} & \quad \text{Gj} \\
Gj & \quad \text{a} \\
Gj & \quad \text{b}
\end{align*}
\]

Figure 3. A bundle relationship

\[
\begin{align*}
S_{ij1} & \quad \text{Gi} \\
S_{ij2} & \quad \text{Gi} \\
S_{ij3} & \quad \text{Gi} \\
Gj & \quad \text{a} \\
Gj & \quad \text{b}
\end{align*}
\]

Figure 4. A path relationship

\[
\begin{align*}
S_{ij1} & \quad \text{Gi} \\
S_{ij2} & \quad \text{Gi} \\
S_{ij3} & \quad \text{Gi} \\
Gj & \quad \text{a} \\
Gj & \quad \text{b} \\
Gk & \quad \text{c}
\end{align*}
\]

The bundle relationship expresses feature variability by grouping alternative features that are mutually exclusive.

The Path relationship: when the achievement of a target goal b from a source goal a requires the satisfaction of intermediary goals, we introduce a path relationship. It establishes a precedence/succession relationship between features expressing that in order to trigger a business flow, some other business flow must be executed first. In general, a path relationship is a composition of features, features related by multi-thread or bundle relationships or other paths. Some paths can be iterative.

Fig. 4 represents a path relationship between the goals Gi and Gj, denoted respectively by a and c, which is composed of the multi-thread relationship containing the features ab1, ab2, ab3 and the feature denoted bc1. It expresses that in order to achieve the goal Gj, we must first select and execute one or many features among ab1, ab2, or ab3, and then execute the feature bc1.

Multi-path relationship: given the multi-thread, bundle and path relationships, a goal can be achieved by several combinations of strategies. This is represented in the map by a pair of goals connected by several sections. Such a relationship is called a multi-path relationship. For example, we show in Fig. 2 two alternative paths to satisfying the goal Gi (denoted c) starting from the goal Gj (denoted a). The first path achieves Gi through the intermediary goal Gj whereas the second path achieves Gi directly from Gj.

A multi-path relationship identifies the several combinations of business flows (represented by paths of sections) that can be executed to satisfy the same goal. Thus, a multi-path relationship is a means to express business feature variability by grouping the alternative paths satisfying the same goal.

Figure 5. A multi-path relationship
In general, a map from its Start to its Stop goals represents all possible combinations of features expressed by multi-thread, multi-path and bundle relationships. Each particular combination of features is a path, from the Start goal to the Stop one, that describes a way to reach the final goal Stop.

2.3 Generating Variants Embedded in a Map

We notice that the bundle and multi-thread relationships are easily visible in the map. However, it is more difficult to identify all the combinations of features in a map (based on multi-path and path relationships). We propose to apply MacNaughton and Yamada’s algorithm (MacNaughton, 1960) in order to discover systematically all the paths embedded in a map. The algorithm is based on the two following formulae:

Let s and t be the source and target goals, Q the set of intermediary goals including s and t and P the set of intermediate goals excluding s and t.

The initial formula $Y_{s,t}$ used to discover the set of all possible paths using the three operators that are the union (“∪”), the composition operator (“.”) and the iteration operator (“∗”) is:

$$Y_{s,t} = (X_{s,qt}, X_{qqt}, X_{qqt}, X_{qt})$$

And given a particular goal q of P, the formula $X_{q,t}$ applied to discover the set of possible paths is:

$$X_{q,t} = X_{q,pt} \cup X_{q,pt} \cup X_{q,pt} \cup X_{q,pt}$$

In this paper we specialize the $X_{q,t}$ into paths, multi-paths, multi-threads and bundle relationships that we note as follows:

- **Multi-thread relationship** between two goals k and l is denoted: $MT_{kl} = \{k \cup l\}$. Where the kl are the features related by the multi-thread relationship. The multi-thread relationship in Fig. 2 is: $MT_{ab} = \{ab\} \cup ab\}$.

- **Bundle relationship** between two goals k and l is denoted: $B_{kl} = \{k \cap l\} \cup \ldots \cup l\}$. Where the kl are the exclusive features related by the bundle relationship. In Fig. 3, the bundle relationship is: $B_{ab} = \{ab\} \cap \{ab\}$.

- **Path relationship** between two goals k and l is denoted $P_{kl}$, where Q designates the set of intermediary goals used to achieve the target goal l from the source goal k. A path relationship is based on the sequential composition operator (“•”) between features and relationships of any kind. As an example, the path relationship of Fig. 4 is denoted: $P_{bc1} = \{bc1\} \cap \{bc1\} \cap \{bc1\}$.

- **Multi-path relationship** between two goals k and l is denoted $MP_{kl}$, where Q designates the set of intermediary goals used to achieve the target goal l from the source one k. A multi-path relationship is based on the union operator (“∪”) between alternative paths. Thus, the multi-path of Fig. 5 is denoted: $MP_{ab} = \{P \cup MT_{a}\} \cup \{P \cup MT_{b}\}$.

In section 3, we illustrate the variability approach with an excerpt of a real case example and show the application of the MacNaughton-Yamada’s algorithm. However, due to paper length limit, we present only the results obtained after applying the algorithm.

It can be seen that the goal of the business process family is captured in a hierarchy of maps. The goal associated to the root map is the high level statement about the purpose of the family. Using the refinement mechanism each section of the root map can be refined as a map and the recursive application of this mechanism results in a map hierarchy. At successive levels of the hierarchy the goal initially as the goal of the root map is further refined. At any given level of the hierarchy, a map describes the business process family as a set of business features and feature variability through four types of feature relationships, namely multi-path, path, thread and bundle relationships. Multi-thread and bundle introduce local variability in the sense that they allow to represent the different ways for achieving a goal directly. Path and multi-path introduce global variability by representing different combinations of business features to achieve a given map goal. Any path from Start to Stop represents one way of achieving the map goal, therefore the purpose represented in this map.

3. AN EXAMPLE

To illustrate our approach, we consider a business example from the ESI (Electricity Supply Industry) sector. The example is simplified to meet the paper size requirement but extracted from a real three-years project involving three large European Electricity companies and our research group (Groso, 1998). The map of Fig. 6 provides the top level intentional view of the electricity supply business family to support Electricity Supply Management (ESM).

The map shown in Fig. 6 is organized around two key goals, “Serve Customer Request” and “Sell Electricity” that represent generic goals in the sense that they exist in any electricity distribution process. Furthermore, the map indicates an ordering constraint: in order to sell electricity to a customer, his/her request for electricity provision has to be fulfilled first.

In the ESM map, it shall be noticed that there are two different strategies to achieve each of these two goals. For example, the “Advance Payment strategy”, and the “Credit strategy”, are two alternative strategies to achieve the business goal “Sell Electricity”. These map strategies identify two rather different business strategies to get the customer to pay for his consumption. Indeed the “Advance Payment strategy” refers to a solution based on the use of payment cards to energise the customer meter whereas the “Credit strategy” refers to the more conventional solution where the electricity company provides electricity to its customer and gets paid after consumption.

Each section in the map represents a feature that the business process family can provide. Further, this example demonstrates the feature variability in the ESM family that is captured by the map. We distinguish two kinds of variability that are:

i. a variability in strategies provided to satisfy the same goal and,
ii. a variability in the combinations of strategies to fulfill the same goal.

The first kind (i) is expressed by the multi-thread or the bundle relationship. In our example, we depict a bundle relationship between the couple of goals “Serve Customer Request” and “Sell Electricity” respectively denoted b and c and composed of the two exclusive features bc1 and bc2 corresponding to the sections <Serve Customer Request, Sell Electricity, Advance payment strategy> and <Serve Customer Request, Sell Electricity, Credit strategy>. We also identify a multi-thread relationship composed of the features ab and bc, corresponding to the sections <Start, Serve Customer Request, Captive strategy> and <Start, Serve Customer Request, Competitive strategy>.

The second kind (ii) is expressed by the multi-path relationship. It shows the different combinations of business flows that can be executed to satisfy the same goal. For example, given an electricity connection obtained after achieving the goal “Serve Customer Request”, we can follow two alternative paths to stop the process. We can either respond to the customer demand applying the “On customer request strategy” or we can proceed with consumed electricity billing and payment.
Managing Worldwide Operations & Communications with Information Technology

675

Copyright © 2007, Idea Group Inc.

Table 1. List business features and their composition

<table>
<thead>
<tr>
<th>Business features</th>
<th>Feature composition kind</th>
</tr>
</thead>
<tbody>
<tr>
<td>ab, ab, bc, bc, cd, bd</td>
<td>Identified compositions</td>
</tr>
</tbody>
</table>

Path  
\[ P_{ab/cd} = MT \lor MP \]

Multi-Path  
\[ P_{bc/ld} = B^* \land cd \]

Bundle  
\[ B_{bc} = bc \lor bc \]

Multi-thread  
\[ MT = ab \lor ab \]

5. CONCLUSION

The notion of variability in business process families introduced here brings together a set of similar but different processes to facilitate reuse and adaptation. We use a goal-driven formalism that is the MAP, to represent business process families as a set of business features and feature variability through four types of feature relationships. Once the process family has been expressed with maps, the task of building the adapted business process model to a given setting can be simply done by deciding which combinations of features are the most suited to the situation at hand. We think that expressing the variability with the map formalism is particularly useful at the adaptation phase. It exposes the business process leader to the choices that are relevant to the satisfaction of her goals in terms of the properties of the business and there is no need to deal with technical configuration details.

This paper has reported the current status of on-going work. Future work consists of (a) implementing a configuration tool to adapt a business process model of a family using the map formalism and (b) developing a software tool to support navigation in a map to select dynamically the feature most appropriate to the situation at hand.

6. REFERENCES


Table 2. Pay-off summary

<table>
<thead>
<tr>
<th>Measure electricity consumption</th>
<th>Get financial counterpart</th>
<th>Contract based</th>
<th>On consumption</th>
</tr>
</thead>
<tbody>
<tr>
<td>Meter reading by meter reader</td>
<td>Can be envisaged at sustainable cost if visits are achieved at a low frequency e.g. once or twice a year</td>
<td>Excluded because too difficult to organise all visits at the required pace.</td>
<td></td>
</tr>
<tr>
<td>Remote reading</td>
<td>Cost effective combination that can be done in real time. However, remote reading is not completely secure. A complementary check of electricity measurement is thus needed, e.g. by meter reader, or by substation inspection.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Semi-remote reading</td>
<td>Cost effectiveness is a linear function of the number of contracts per cluster of semi-remote reader.</td>
<td>Very costly if the number of customers paying on consumption, per cluster of remote reader is low.</td>
<td></td>
</tr>
<tr>
<td>Substation inspection</td>
<td>Only possible if the connected meter readers relate to single contract. Otherwise, calls for individual reading.</td>
<td>Cost effective way to handle the verification of consumers invoiced by remote reading clustered on the same substation.</td>
<td></td>
</tr>
</tbody>
</table>


Related Content

Probability Based Most Informative Gene Selection From Microarray Data
www.irma-international.org/article/probability-based-most-informative-gene-selection-from-microarray-data/190887

Corporate Social Responsibility
www.irma-international.org/chapter/corporate-social-responsibility/183780

A Short Review of Fuzzy Linear Equation Systems
www.irma-international.org/chapter/a-short-review-of-fuzzy-linear-equation-systems/112396

An Efficient Image Retrieval Based on Fusion of Fast Features and Query Image Classification
www.irma-international.org/article/an-efficient-image-retrieval-based-on-fusion-of-fast-features-and-query-image-classification/169172

Illness Narrative Complexity in Right and Left-Hemisphere Lesions
Umberto Giani, Carmine Garzillo, Brankica Pavic and Maria Piscitelli (2016). *International Journal of Rough Sets and Data Analysis* (pp. 36-54).
www.irma-international.org/article/illness-narrative-complexity-in-right-and-left-hemisphere-lesions/144705