



This paper appears in *Managing Modern Organizations Through Information Technology*, Proceedings of the 2005 Information Resources Management Association International Conference, edited by Mehdi Khosrow-Pour. Copyright 2005, Idea Group Inc.

Conceptual Optimization in Business Process Management

Yves Callejas, Jean Louis Cavarero and Martine Collard

Université de Nice-Sophia Antipolis, Laboratoire I3S, 2000, route des lucioles, 06903 Sophia Antipolis cedex, France

callejas@wanadoo.fr, jlcava@aol.com, mcollard@unice.fr

ABSTRACT

To optimise business processes is a very complex task. The goal is double: to improve productivity and quality. The method, developed in this paper, is composed of 4 steps : the first one is the **modelisation step** (to describe the business process in a very rigorous way), then a **conceptual optimisation** (supported by evaluation and simulation tools) to improve the business process structure (to make it more consistent, to normalise it), then an **operational optimisation** to improve the business process performing (to make it more efficient) by providing to each operation the necessary resources and at last a **global optimisation** (to take into account all the business processes of the company under study). The conceptual optimisation is, in fact, a static optimisation (achieved independently of resources) while the operational optimisation is dynamic. The main difference between these 2 steps is the fact that the first one is totally hand made (we want to build, from the set of indicators provided by evaluation and simulation, the best business process as possible), in opposition with the second which is totally automatic (since it requires linear and non linear programming tools).

This method is the result of three years research achieved for the French organism "Caisses d'Allocations Familiales: CAF". It was validated on the business processes of the CAF, which deal with information (files and documents), but it can also be applied on industrial business processes (dealing with products and materials).

INTRODUCTION

Business process optimisation is one of the major issues of any company. The main goals are to improve processes quality and to improve their productivity, by increasing the number of output flows and/or by decreasing the quantity of necessary resources.

Since the appearing of object oriented paradigm, methods to design Information Systems have been oriented towards the class notion instead of the process notion. But evolutions of technical environment, use of Internet technology, led to consider Information Systems as a tool to change ways of working and led companies to adopt a process approach. A **business process** (BP) is composed of **operations** (tasks) (automated or not) and **resources** (possessing **competencies**) which have to deal with **flows** (of data, or materials). Business process **management** and **optimization** are quite recent ideas since processes were implicit in each organisation culture so far.

The method presented in this paper starts with a previous modelling step followed by three main steps:

- Modelisation step makes it possible to represent BP with a model which has the usual guarantees of any good model: **readability**, **normalisation**, **genericity**, and which induces an optimisation more rigorous, more consistent and less hazardous. Modelisation was decided for all these reasons in order to avoid an empirical optimisation consisting in improving each BP from clues based on its behaviour, by trying to find out **local** solutions for the parts which run worst or the less correctly.
- Conceptual optimisation step which does not take into account resources; it is a structural and static optimisation.

- Operational optimisation step which consists in optimising the performing of the BP by taking into account resources, which means by locating them the best way as possible. It is a dynamic optimisation since the goal is to optimise performances.
- Multi-BP optimisation step which is used to optimise (in the operational way) several BP simultaneously.

MODELISATION

Concepts

Four concepts are necessary to model business processes: **operation, flow, resources and competencies**.

- **OPERATIONS**: an operation is a task of a business process. Each operation can be mandatory or optional, and disactivable (or not). A mandatory operation has to be used systematically (always), which means it is necessary for the right performing of the BP. An optional operation may not be used depending on the decided options. A disactivable operation is an operation which can be shunted if the corresponding resources are not available.
- **FLOWS**: A flow is a set of homogeneous elements passing through the BP and treated by operations. An optional flow is a flow associated to an optional operation. We consider 3 types of flows: primary flows, secondary flows and triggering operations events.
 - Primary flows : are the flows (of data or materials) which are to be treated (in input or output)
 - Secondary flows: are data flows containing useful and helpful information to deal with primary flows (guidelines, precisions, complements,...)
 - Triggering operations events: are necessary events (not always sufficient because a primary flow is most of the time also necessary) to trigger operations (for example a special date).
- **RESOURCES AND COMPETENCIES**: These 2 concepts are linked. A resource is a group of persons having the same set of competencies. A resource possesses one or several competencies. A competency can be associated to several resources (N: M link). The set of resources is a partition of the persons set.

We consider that each operation is one-competency (to be performed an operation needs persons having the same competency). In a BP, each swim lane contains all the operations which need the same competency. A specific swim lane is associated to automatic operations.

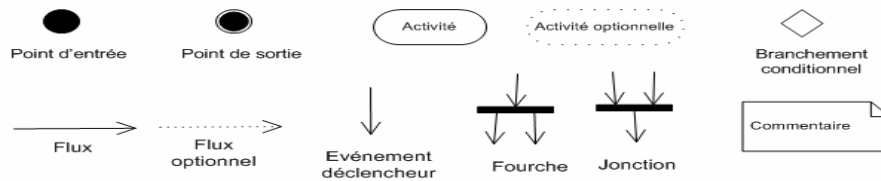
Model

The chosen model is directly inspired from the UML activity diagrams. Fig. 1 shows the UML concepts which are used. It is a subset of the available set of elements in UML.

Tool

In order to build activity diagrams, we can use any tool supporting UML notations. We can choose either a full UML environment like RATIO-

Figure 1. UML Concepts to Build Activity Diagrams



NAL ROSE or DESCRIBE or a graphic modelling tool like VISIO or SMARTDRAW. Several criteria are to be taken into account to make the right choice: conformity with the UML 1.4 norm, graphic quality of documents, opportunity of exporting diagrams in XML in order to provide right inputs for simulation tools, learning facilities, ergonomics.

DESCRIBE was finally chosen, because it is the tool which satisfy the better these criteria. As an example, Fig 2 shows a part of a real diagram built for the CAF.

CONCEPTUAL OPTIMISATION

The modelling step provides a diagram of the BP (by using the previous concepts) in order to **evaluate** it (with simulation and evaluation tools) and to **optimise** it in the right directions by taking into account its structure and its defaults (and still so far without resources). The conceptual optimisation is static.

Evaluation Step

The goal of this step is to collect all useful information to improve the BP structure. This information is given by indicators and objectives graph. Some of them are provided by simulation.

Indicators

Indicators are used to evaluate a BP. They are of 2 kinds: model indicators and BP indicators.

- Model indicators: they are used to evaluate the consistency of a BP independently of its finality. They are theoretical indicators (in opposition to BP indicators). They provide an evaluation of the diagram quality and make it possible to check that diagrams are satisfying the norms given by the model. In others words, to check that the conceptual optimisation step delivers **well built** diagrams. The list of model indicators is following: maximum number of input flows in each operation, maximum number of output flows in each operation, average number of flows per operation, number of optional operations / number of operations, number of disactivable operations / number of operations, number of loops, cyclomatic number (number of bows-number of nodes + 2), diagram density (number of bows/maximum number of bows), average number of operations per competency, average number of flows per competency.
- BP indicators: they are used to evaluate performances and dysfunctionnings of a BP. Their values are useful to determine the optimisation priorities (see objectives graph, Fig. 4).

Once achieved the conceptual optimisation, a second evaluation of these indicators can be done in order to check that the goals have been satisfied. The modelling step and the objectives graph step make it possible to find out (for a given BP) the list of the useful indicators.

Fig. 3 shows some examples of BP indicators in a specific BP from the CAF. These indicators are located in the graph (see Fig. 4).

Hierarchical Objectives Graph

It is necessary to build a hierarchy of optimisation objectives and to identify precisely those which are means compare to the others. Thus, we propose to build a “hierarchical objectives graph” (HOG). This kind of graph makes it possible to show clearly the hierarchical relationships between objectives.

If the graph is well built and exhaustive, all its leaves **are the actions to perform** in order to optimise the BP. More precisely, the graph is built by connecting (if possible) to each node (objective) some indicators, values of which will be provided by evaluation and simulation steps (in the example I1, I2 and I3). The graph is helpful to build an optimise BP because it gives the hierarchical links between objectives and then optimisation priorities. Each BP has its own graph. The bows of the graph have to be valuated (with percentages) in order to give the satisfaction weight of an objective to another one (higher in the hierarchy) and to guide the process optimisation.

Fig. 4 shows a part of the objectives graph of the BP introduced in Fig. 3.

Figure 2. Example of Business Process Diagram Built with DESCRIBE

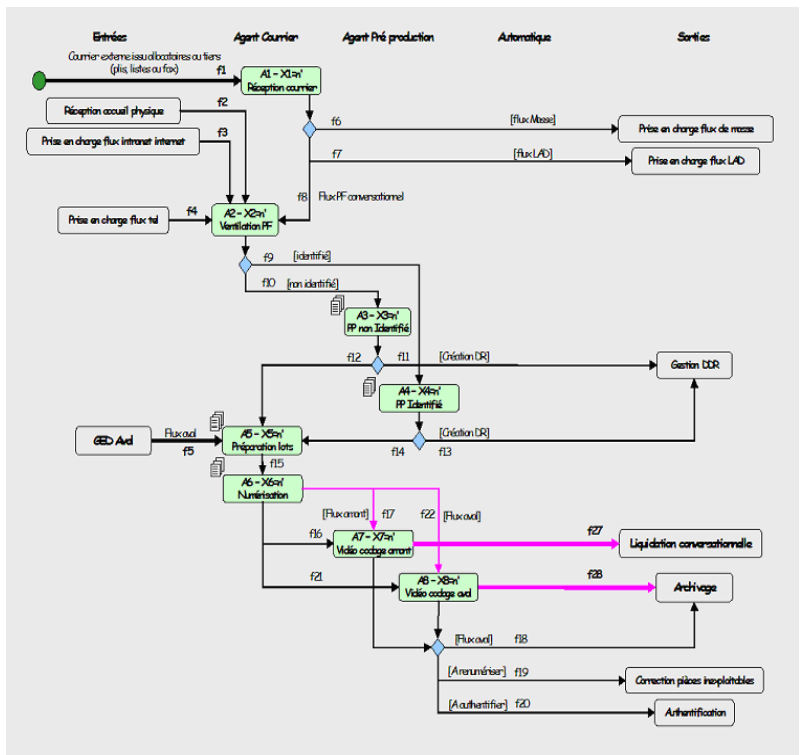
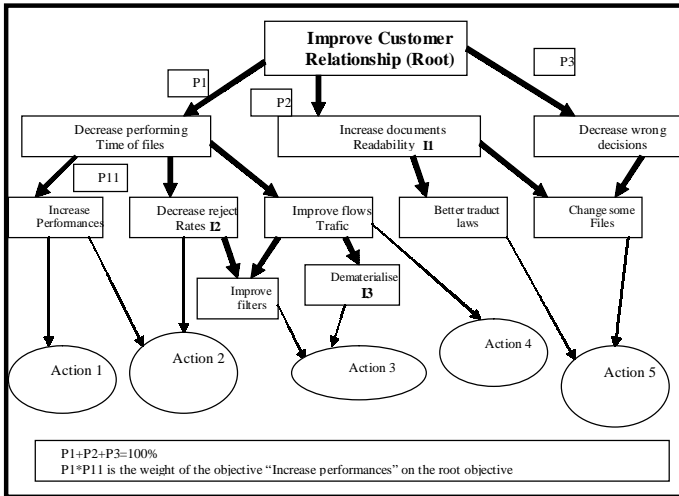


Figure 3. Examples of BP Indicators

NODES OF THE GRAPH	INDICATORS	FORMAT	OBTENTION
READABILITY OF DOCUMENTS	I1 :%SATISFACTION PER DOCUMENT	VECTOR	OBSERVATION
DECREASE RATE REJECT	I2 :% REJECT PER OPERATION	VECTOR	SIMULATION
DEMATERIALIZATION	I3 :% DEMATERIALIZATION	NUMBER	OBSERVATION

Figure 4. Example of Hierarchical Objectives Graph



Simulation Step

This step is dedicated to the study of the BP behaviour in order to find out some of the possible improvements (addition or deleting operations and/or flows, detection of wrong cycles, detection of congestion points,...). Obviously, this step requires a simulation tool (SIMPROCESS was chosen).

The simulation step is also used to give values to indicators, such as the reject ratio per operation. It is an essential step to evaluate correctly BP.

Note: the use of a simulation tool requires to modelise the BP (in the form wanted by the tool). In our case, since BP are already modelled (in DESCRIBE), we had to build a specific tool to convert DESCRIBE diagrams into SIMPROCESS diagrams (in XML).

Conceptual Optimisation Step

The conceptual optimisation of a BP is achieved from information provided by evaluation step, simulation step and objectives graph step. The goal is to build the best BP as possible (in regards to norms, indicators, objectives hierarchy). It is a very tough step (totally hand made) which requires to take into account simultaneously a very large number of information and a great know how. Thus, values of some model indicators will induce creation or suppression of some operations and/or flows, values of some BP indicators will generate creation of some new paths in the diagram (by validating or deleting optional operations) or creation of new documents, analysis of the objectives graph make it possible to identify the parts of the BP which have to be optimised in priority.

Conceptual optimisation is totally guided by the objectives graph: weights are used to know priorities and indicators are used to decide if the nodes are easy to optimise or not. In the example, we can decide to give a priority to the objective "decrease the time to perform a file" if the values of I2 are too high and if the weight of this objective in regards to the root objective is high. In this case, we have to (following the graph) modify some resources and add some operations. In opposition, if the value of I1 is too low and if the weight of the objective "to increase readability of documents" is high, then we have to design new documents.

Actually, the conceptual optimisation of a BP is achieved par a lot of improvements (defined in the leaves of the graph) performed on its diagram, in regards to the objectives graph which gives the right directions. But the diagram's improvement has to be done in respect of concepts: we can't do anything forbidden or in opposition with the given rules. For this reason, we have defined the exhaustive list of generic actions (meta-actions) which are possible to do. This list is the tools box in which the designer can find any action which may optimise the BP. **Each leaf of the graph has to be obviously an instance of one meta-action.**

Examples of meta-actions: to add a new operation, to automate partially an operation, to split an operation (in 2 or more) to add a new flow, to split a flow (in 2 or more), to merge 2 or more flows into 1, to modify a flow, to add a new competency, to modify a competency's profile, to change the destination operation of a flow.

Fig. 5 shows the whole schema of a conceptual optimisation. We may note that evaluation and simulation can be performed several times: once to provide necessary information to optimise, and then others to check the results of the optimisation.

OPERATIONAL OPTIMISATION

This step consists in giving to each operation of a BP, resources and competencies, in order to **maximise** output flows. Actually, the final goal is to provide a **command tool** to predict the best resources affectation as possible, by taking into account different hypothesis of degraded performing (for example absenteeism) as well as flows stocks (flows which have not been treated).

This third step is divided in two distinct issues:

- Issue 1:** Searching optimum of outputs flows (by an optimised affectation of resources and competencies to operations (**linear optimisation**)).
- Issue 2:** Locating resources and competencies on each operation **at the right time (non linear optimisation)**.

To illustrate this step, let's take an example of BP given in Fig. 6.

This BP is composed of 8 operations (A1, A2,..., A8) and 9 resources (R1, R2,..., R9). The relationships between resources and competencies are given in Fig. 7, the stocks of flows are given in Fig. 8.

Let f_p be the number of units of flow p treated and w_{jk} be the used time of the resource R_j for its competency C_k (during the chosen period).

Issue 1 consists in giving resources and competencies to each operation (by finding out optimal values of f_p and w_{jk}) (who does what?) and in computing the maximum number of output flow units (which can be treated on the period).

The equations system to solve is illustrated in Box 1; it corresponds to using conditions of competencies on the period.

The first legs of these equations are the total used time of competencies. For example, competency C1 which is used in operations A1 and A3 is engaged for a time $T1*f3$ in A1 and $T3*f9$ in A3. The second legs correspond to the used time of competencies in regards to resources. For example, competency C1 is provided for a time $w21$ by resource R2, for a time $w41$ by resource R4 and for a time $w91$ by resource R9.

These equations can also be formulated by using variables x_i (instead of variables f_p), x_i being the number of times operation A_i is performed. Fig. 9 shows relationships between x_i and f_p .

Box 1.

Competency C1	$T1*f3 + T3*f9 = w21 + w41 + w91$
Competency C2	$T2*f6 + T5*(f4+f5) + T7*f10 = w12 + w22 + w62 + w72 + w82$
Competency C3	$T6*f8 = w33 + w63$
Competency C4	$T4*f7 + T8*f11 = w44 + w54$

Box 2.

Availability constraints of resources:

For each resource, total used time should not be higher than available time. There are 9 constraints of this kind:

- Resource R1: $w12 \leq TR1$
- Resource R2: $w21 + w22 \leq TR2$
- Resource R3: $w33 \leq TR3$
- Resource R4: $w41 + w44 \leq TR4$
- Resource R5: $w54 \leq TR5$
- Resource R6: $w62 + w63 \leq TR6$
- Resource R7: $w72 \leq TR7$
- Resource R8: $w82 \leq TR8$
- Resource R9: $w91 \leq TR9$

Pouring constraints on flows: Input and output flows can be multiple.

To express that input flow $f2$ is appreciatively 60% of the total input flow ($f1+f2$), we write one constraint on the flow from $f1$ and on the flow from $f2$: $f2 \geq 1.4 * f1$ and $f2 \leq 1.6 * f1$.

To express that output flow $f4$ is appreciatively 80% of the total output flow ($f4+f5$), we write one constraint on the flow from $f4$ and on the flow from $f5$; we write a similar constraint on the flows $f7$ and $f8$: $f4 \geq 3.9 * f5$ and $f4 \leq 4.1 * f5 / f7 \geq 3.9 * f8$ and $f7 \leq 4.1 * f8$.

For each operation, the sum of output flows has to be inferior to the sum of input flows:

$$f11 \leq f9 + f10, f9 \leq f6, f6 \leq f3, f3 \leq f1, f10 \leq f7 + f8, f7 \leq f4, f8 \leq f5, f4 + f5 \leq f2$$

Figure 5. The Conceptual Optimisation Step

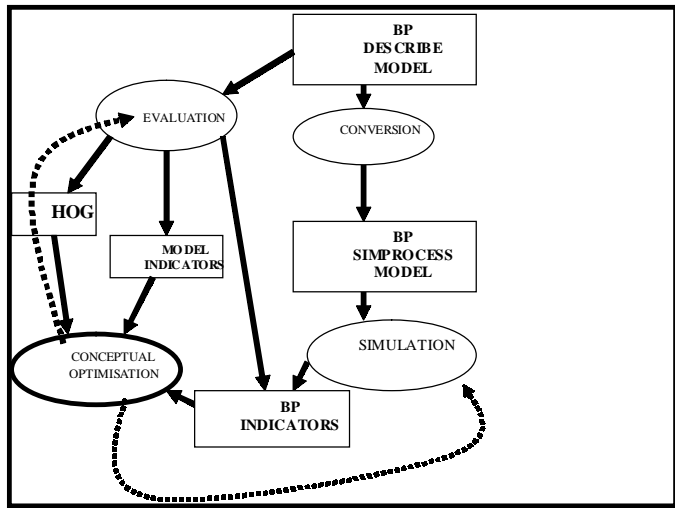
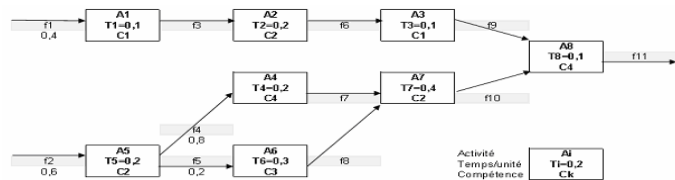


Figure 6. Example of BP



To solve the system we also have to take into account two types of constraints (see Box 2).

These equations can also be formulated by using variables x_i (instead of variables f_p), x_i being the number of times operation A_i is performed. Fig. 9 shows relationships between x_i and f_p .

The solver provides values of x_i , f_p and w_{jk} which optimise output flows. The results are given in the next tables.

Issue 2 consists in searching dated resources locations. (Who does what and when?).

For this reason, we have to split the period (35 hours in the example) in 10 slices of same length D (3,5 h in the example), and we have to find out quantities of resources to give to each operation in each slice.

The result will be the used resources and competencies for each slice.

For this issue, the system solving has to take into account 3 types of constraints:

Figure 7. Table Resources/Competencies

RESOURCE	COMPETENCY	AVAILABILITY/PERIOD
R1	C2	TR1=35
R2	C1, C2	TR2=35
R3	C3	TR3=35
R4	C1, C4	TR4=35
R5	C4	TR5=35
R6	C2, C3	TR6=35
R7	C2	TR7=35
R8	C2	TR8=17.5
R9	C1	TR9=17.5

Figure 8. Table of Flows Stocks

FLOWS	STOCKS
f1	29
f2	58
f3	6
f4	2
f5	6
f6	4
f7	4
f8	2
f9	4
f10	8
f11	0

Figure 9. Table x_i/f_p

x_i	f_p
x1	f3
x2	f6
x3	f9
x4	f7
x5	f4+f5
x6	f8
x7	f10
x8	f11

Exclusivity constraints on competencies:

For each multiple competency resource, at most one competency is used in each slice. As we have 9 resources and 10 slices, we have 90 constraints of this kind. If we name $\{C_{jk}\}_{k \in 1..p}$ the set of competencies associated to resource R_j , the constraint may be expressed in the following way:

$$\forall (\text{Resource } R_j, \text{ slice } t) \exists \text{ at most one } k \in 1..p \text{ such that } C_{jk} \text{ is used in slice } t.$$

Figure 10. Table of Competencies Used Time

RESOURCE		$\Sigma_k w_{jk}$	AVAILABLE TIME
R1	w12 : 28	28	7
R2	w21 : 10.5 w22 : 21	31.5	3.5
R3	w33 : 17.5	17.5	17.5
R4	w41 : 3.5 w44 : 31.5	35	0
R5	w54 : 21	21	14
R6	w62 : 28 w63 : 3.5	31.5	3.5
R7	w72 : 31.5	31.5	3.5
R8	w82 : 17.5	17.5	0
R9	w91 : 17.5	17.5	0
TOTAL		231	49

Figure 11. Flows Results

Input flows	
f1	95
f2	176
Stocks	
Total (input + stocks)	394
Output flows (f11)	280
Non treated flows	114

Figure 12. Table of Operations Runs

OPERATION	NUMBER OF RUNS	TOTAL PERFORMING TIME
A1	x1 96	10.5
A2	x2 102	21
A3	x3 35.6	21
A4	x4 120.4	24.5
A5	x5 148	38.5
A6	x6 35.6	21
A7	x7 162	66.5
A8	x8 280	28

Figure 13. Table of Operations Availability

COMPETENCY Ck	POTENTIAL Pck	USED Uck	AVAILABLE Dck
C1	87.5	31.5	3.5
C2	157.5	126	17.5
C3	70	21	21
C4	70	52.5	14
TOTAL	385	231	56

Using constraints of resources in operations:

They are equality constraints. For each slice and for each competency, there is equality between quantities of competencies used by operations and quantities of competencies taken in resources. In the example, there are four competencies and 10 slices; we have then 40 constraints of this kind.

If we name $\{aik(t)\}_{k \in 1..p}$ the set of used times of competency Ck for the operation Ai on the slice t and $\{w'jk(t)\}_{k \in 1..q}$ the set of used times of resource Rj for its competency Ck on the same slice, the constraint may be expressed in the following way: $\Sigma_i aik(t) = \Sigma_j w'jk(t)$

Where $aik(t) = \alpha ik(t) * D$; $\alpha ik(t)$ representing the number of times competency Ck is used for operation Ai on the slice t (that is the number of used resources).

Evolution constraints on flows:

We assume that flows evolve in a discontinuous way. After each slice, flows evolve in regards to resources provided to operations and available flows of the previous slice. Let's take a basic example (Figure 14).

The formula which gives flow fb after slice t is:

$$Flux(b,t) = Flux(b,t-1) + Min(Flux(a,t-1), aik(t)*D/Ti) - Min(Flux(b,t-1), ajk(t)*D/Tj)$$

Where $\alpha ik(t)$ represents the number of times competency Ck is used for operation Ai on the slice t.

It is necessary to adapt this formula if operation Ai or Aj are preceded and/or followed by several operations.

An additional table is used to give quantities of input flows on each slice (in regards to the chosen arrival law).

The solver provides values of aik(t) which optimise the repartition of resources and competencies on each slice and associated flows.

The table of Fig. 16 shows, the number of times aik(t) competency Ck is used for operation Ai for the slice t. For example, on the first slice, three resources of competency C2 are given to operation A5 and on the fourth slice two resources of competency C3 are given to operation A6.

The table of Fig. 17 shows the number of flows units treated after each slice.

The table of Fig. 18 shows, for each resource and each slice, the use of associated competencies. For example, on the first slice, resources R2, R6 and R7 are the three resources of competency C2 given to operation A5 and on the fourth slice, resources R4 and R5 are the two resources of competency C4 given to operation A6.

The last three tables give the command diagram of the BP which makes it possible to pilot by defining for each slice, the optimal affectation of resources and competencies to operations.

Note: the tool « Premium Solver » was used for the operational step.

MULTI-BP OPTIMISATION

As indicated by its name, multi-BP optimisation consists in optimising simultaneously several BP. Obviously, this step does not involve conceptual optimisation (which is, by definition, made on one BP,

Figure 14. Evolution of Flows

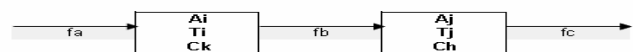


Figure 15. Table of Input Flows

during	slice1	slice2	slice3	slice4	slice5	slice6	slice7	slice8	slice9	slice10
input f1	15	21	15	8	8	8	8	6	4	2
input f2	32	58	14	14	12	11	12	10	10	3

Figure 16. Table of Competency/Operation Affectation for Each Slice

OPERATION	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	COMPETENCY
A1	1	0	1	0	0	1	0	0	0	0	C1
A2	0	1	0	0	0	1	3	1	0	0	C2
A3	0	0	0	0	1	1	0	2	2	0	C1
A4	0	1	1	1	1	2	1	0	0	0	C4
A5	3	3	5	0	0	0	0	0	0	0	C2
A6	0	0	1	2	1	1	0	1	0	0	C3
A7	0	0	0	4	5	3	1	3	3	0	C2
A8	0	0	0	0	1	0	1	2	2	2	C4

Figure 17. Table of Flows Units Treated After Each Slice

TREATED FLOWS UNITS	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10
f1	15	36	16	24	32	8	16	22	26	28
f2	38	58	14	28	40	51	63	73	83	86
f3	35	18	53	53	53	67	15	0	0	0
f4	44	57	85	68	50	15	0	0	0	0
f5	17	24	24	1	0	0	0	0	0	0
f6	4	22	22	22	0	18	70	15	0	0
f7	4	22	39	31	25	40	47	21	0	0
f8	2	2	14	28	8	2	1	1	0	0
f9	4	4	4	4	23	23	15	71	38	0
f10	8	8	8	43	55	81	62	32	32	0
f11	0	0	0	0	35	35	70	140	210	280

Figure 18. Table of Used Competencies and Resources on Each Slice

RESOURCE	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10						
R1 (C2)	0	0	1	0	1	0	1	0	1	0	1	0	1	0	0	0
R2 (C1, C2)	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	0
R3 (C3)	0	0	0	1	0	1	0	1	0	1	0	1	0	1	0	0
R4 (C1, C4)	0	0	0	0	1	0	1	0	1	0	1	0	1	0	1	0
R5 (C4)	0	0	0	0	0	0	1	0	1	0	1	0	1	0	1	0
R6 (C2, C3)	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
R7 (C2)	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
R8 (C2)	0	0	0	0	1	0	1	0	1	0	1	0	1	0	1	0
R9 (C1)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

independently from the others) but only operational optimisation in the case where resources and competencies are shared by several BP.

If so, the persons in charge of the business processes have to define:

1. priorities between BPs and between objectives
2. constraints on resources and competencies which will be given to BPs and operations

For point 1, priorities will be listed in a decreasing order (example: for BP 1 / objective: to get 90% of input flows in output, BP 3 / objective: to get 75% of input flows in output,...).

For point 2, constraints will also be listed in a decreasing order. These constraints are of two kinds:

- To give a maximum quantity Q of resource R of competency C to one particular BP or to one particular operation (example: to give 20 units maximum of resource R2 to operation O3).
- To give in priority resource R or competency C to one particular BP or to one particular operation (example: competency C4 is dedicated in priority to BP2).

When priorities and constraints are defined, the solver can be run (one to N times) by deleting one by one constraints (from the bottom of the list) while objectives are not satisfied. The multi-BP optimisation step is, thus, a generalisation of the operational optimisation step, using the same tool and being done several times in a row. The final goal is to achieve a full and global command of all the BPs of the company.

CONCLUSION

The optimisation method presented in this paper is composed of 4 steps: modelisation step, conceptual optimisation step, operational optimisation step, multi-BP optimisation step. Its originality consists

in separating clearly issues related to modelisation and issues connected to optimisation. The first step (modelisation step) is necessary to model BPs under study and so necessary for the 3 others steps. The second one (conceptual optimisation step) make it possible to build the best BPs as possible, consistent and normalised (in regards to norms, objectives and indicators). It is an optional step, since it is not necessary to achieve operational optimisation. It is, though, strongly recommended because it is not very judicious to try to affect resources and competencies on a BP which is not correctly built. The third one (operational optimisation) is probably the main one. Its goal is to improve the performances and behaviour of BPs by optimising resources and competencies locations. The main advantage of this 2 steps optimisation is to improve BPs quality as well as better control their evolution.

This method was validated on administrative BPs. It also works on industrial BPs, under condition to take into account (during the operational optimisation) issues of breakdowns and maintenance of machines (by using complementary tools), issues which were not presented in this paper. This research is going to be extended by introducing data mining techniques in the conceptual step in order to find out more efficient optimising rules. We would like to thank the CNEDI 06 and more particularly M.P. Bourgeot who made this research possible.

BIBLIOGRAPHY

[Aler 2002a] Ricardo Aler, Daniel Borrajo, David Camacho, and Almudena Sierra-Alonso. A knowledge-based approach for business process reengineering, SHAMASH. Knowledge Based Systems, 15(8):473–483, 2002.

[Aler 2002b] Ricardo Aler, Daniel Borrajo, and Pedro Isasi. Using genetic programming to learn and improve control knowledge. Artificial Intelligence, 141(1-2):29–56, October 2002.

[Ambite 2001] Jose Luis Ambite and Craig A. Knoblock. Planning by rewriting. Journal of Artificial Intelligence Research, 15:207–261, 2001.

[Borrajo 2001] Daniel Borrajo, Sira Vegas, and Manuela Veloso. Quality based learning for planning. In Working notes of the IJCAI’01 Workshop on Planning with Resources, pages 9–17, Seattle, WA (USA), August 2001, IJCAI Press.

[Estlin 1996] Tara A. Estlin and Raymond J. Mooney. Multistrategy learning of search control for partial-order planning. In Proceedings of the Thirteenth National Conference on Artificial Intelligence, volume I, pages 843–848, Portland, Oregon, August 1996. AAAI Press/MIT Press.

[Iwamoto 1994] Masahiko Iwamoto. A planner with quality goal and its speedup learning for optimization problem. In K. Hammond, editor, Proceedings of the Second International Conference on Artificial Intelligence Planning Systems (AIPS94), pages 281–286, Chicago, Illinois, June 1994.

[Jensen 2001] Paul A. Jensen, Jonathan F. Bard. Operations Research Models and Methods, ISBN : 0-471-38004-0, Wiley Higher Education

[Krajewski 2001] Lee J. Krajewski, Larry P. Ritzman. Operations Management: Strategy and Analysis (6th Edition), Prentice Hall

[Nareyek 2001] Alexander Nareyek, editor. Working notes of the IJCAI’01 Workshop on Planning with Resources, Seattle, WA (USA), August 2001. IJCAI Press.

[Williamson 1994] Mike Williamson and Steve Hanks. Optimal planning with a goal-directed utility model. In K. Hammond, editor, Proceedings of the Second International Conference on Artificial Intelligence Planning Systems (AIPS94), pages 176–181, Chicago, Illinois, June 1994.

0 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/proceeding-paper/conceptual-optimization-business-process-management/32582

Related Content

Detection of Automobile Insurance Fraud Using Feature Selection and Data Mining Techniques

Sharmila Subudhi and Suvasini Panigrahi (2018). *International Journal of Rough Sets and Data Analysis* (pp. 1-20).

www.irma-international.org/article/detection-of-automobile-insurance-fraud-using-feature-selection-and-data-mining-techniques/206874

Taxonomy of IT Intangible Assets for Public Administration Based on the Electronic Government Maturity Model in Uruguay

Helena Garbarino, Bruno Delgado and José Carrillo (2013). *Information Systems Research and Exploring Social Artifacts: Approaches and Methodologies* (pp. 103-122).

www.irma-international.org/chapter/taxonomy-intangible-assets-public-administration/70712

Sustainability

Yannis A. Phillis (2015). *Encyclopedia of Information Science and Technology, Third Edition* (pp. 6935-6947).

www.irma-international.org/chapter/sustainability/113163

Weighted SVMBoost based Hybrid Rule Extraction Methods for Software Defect Prediction

Jhansi Lakshmi Potharlanka and Maruthi Padmaja Turumella (2019). *International Journal of Rough Sets and Data Analysis* (pp. 51-60).

www.irma-international.org/article/weighted-svmboost-based-hybrid-rule-extraction-methods-for-software-defect-prediction/233597

Knowledge Visualization for Research Design: The Case of the Idea Puzzle Software at the University of Auckland

Ricardo Morais and Ian Brailsford (2019). *Enhancing the Role of ICT in Doctoral Research Processes* (pp. 46-66).

www.irma-international.org/chapter/knowledge-visualization-for-research-design/219931