Available online at www.sciencedirect.com

ScienceDirect

Procedia Technology 9 (2013) 388 – 398

Procedia Technology

CENTERIS 2013 - Conference on ENTERprise Information Systems / PRojMAN 2013 - International Conference on Project MANagement / HCIST 2013 - International Conference on Health and Social Care Information Systems and Technologies

Goal-Driven Approach For Business/IT Alignment Evaluation

Lerina Aversano*, Carmine Grasso, Maria Tortorella

Departement of Engineering, University of Sannio
Via Traiano, 82100 Benevento, Italy

Abstract

The alignment degree existing between a business process and the supporting software systems strongly affects the performance of the business process execution. Methodologies and tools are needed for detecting the alignment level and either keeping a business process aligned with the supporting software systems even when they evolve or increasing its level by identifying convenient software changes. This paper improves an approach previously proposed for managing alignment. In particular, a new set of metrics have been introduced in the existing approach for business goals and dynamic behavior to evaluate the alignment. The application of the new approach is explored through a case study.

Keywords: Business/IT Alignment; Business Process Modelling; Software System Modelling; Business Process Evaluation; Software System Evaluation.

1. Introduction

The issue of alignment between business processes and involved software systems is considered as a relevant challenge from business executives and information technology practitioners [7]. Basically the alignment refers to the degree of fit between business needs and the support provided by software systems. The

* Corresponding author. Tel.:+39 0824 305551; fax: +39 0824 50552
*E-mail address:*aversano@unisannio.it.

doi:10.1016/j.protcy.2013.12.043

performance of a business process is interrelated with the adequacy of software systems. Software aims to provide services at all levels of a business organization to effectively achieve their goals and objectives [5] [12]. However, achieving strong alignment remains a difficult process for enterprises due to a lack of adequate support and rapid changes of the business environment, mainly with reference to changes in consumer services, technologies and product lifecycles.

A precise definition of the alignment concept lacks, even if the concept is clear. Generally, it is described at two different abstraction levels, i.e. strategic and functional [7] [11], and involves different concepts, such as enterprise goals, business entities, strategies and processes, technology, information system and data. In particular, strategic alignment of IT exists when goals, processes and activities of a business organization are in harmony with the information systems supporting them. The functional level regards the alignment existing between business processes and software systems and aims at optimizing the effectiveness of the software support during the business process execution.

This paper focuses on the functional level of alignment and proposes an improvement of an existing approach for measuring the alignment. The enhancement introduced focus of the high level specification of users needs.

The aim is to improve the ALBIS (ALigning Business Processes and Information Systems) strategy for evaluating the alignment existing between a business process and related system systems [1]. The improvement considers the introduction of an appropriate understanding of the business process goals.

The identification of business goals drives the establishment of an agreement between business objective and software systems functionalities. Their specification can be used to improve the assessment of the software system behaviour and adequacy of its support.

Business goals are typically expressed trough an enterprise language. In this paper the goals are defined and formalized with reference to the dynamic issues of business process and artefact entities.

Numerous modelling techniques are described in the literature, but they do not consider business goals in the context of business/IT alignment [6] [8] [9] [10]. They also appear to be limited to support business analysts in the rapidly changing business environment, as they do not provide any operative support for being applied and require a huge amount of time to be implemented [3].

This paper presents a business goal-driven alignment approach in the context of information technology (IT) alignment and provides the following two contributes:

- Consider the alignment of software systems with the enterprise business goals, extending an ALBIS evaluation strategy already proposed by the authors [1];
- Analyse to what extent considering a business goal influences the alignment existing between a business process and software systems;

The rest of the paper is organized as it follows: Section 2 discusses the extension of the ALBIS strategy; Section 3 describes the application of the approach to a case study; finally conclusions are given in the last Section 4.

2. Alignment Evaluation

The assessment of the alignment level existing between a business process and the supporting software systems entails the evaluations of quantitative information. This requires the identification of suitable metrics. The approach proposed in [1][2] considered quality attributes such as the Technological Coverage and Technological Adequacy and a set of related metrics considering technological coverage and adequacy of actors, activities and artefacts. In this paper, this approach is reviewed and enhanced for introducing additional quantitative measures regarding the alignment with reference to the business goal satisfaction.

The measurement framework of the ALBIS approach, supporting the assessment of the alignment degree, is defined on the basis of the Goal Question Metrics (GQM) paradigm, defined by Victor Basili [4]. The use of this paradigm starts from the definition of the evaluation goals, and analyses the questions to be answered for reaching the stated goals and metrics to be assessed for answering the questions.

In the analysed context, the measurement goal can be formulated as follows:

Analyse a business process and the supporting software systems with the aim of evaluating the alignment level existing between them from the point of view of the software engineer and business analyst.

The hierarchical structure of the GQM model entails the two major components: Technological Coverage and Technological Adequacy. Each one is refined into metrics that are brought to the essential aspects involved in a process model, classified in activities and resources involved in the activity execution, such as human resources (actors), and input/output data required for performing the activities (artefacts). The term artefact is used for information considered in the business process at any granularity level.

The metrics defined in the ALBIS approach evaluated the alignment characteristics by considering static aspects of the business process. In addition to them, the current proposal introduces further metrics specifically defined to consider dynamic aspects concerning the achievement of the business goal in terms of technical coverage and adequacy. A business goal is a set of stable states that have to be valid after the business process is executed. A goal of a process is represented in an activity diagram and formulated as a Boolean function concerning the states of a set of artefacts involved in a process and other states that have to be reached at the end of the process. Examples of goals can be found in Table 1. They regard the analysed case study and related business processes and are formulated combining the initial and final states of the artefacts.

With reference to the technological coverage the measurement framework analyses the coverage level for each essential aspect of a business process. In particular, both static and dynamic aspects are considered.

The metrics regarding the static aspects and defined in the previous version of ALBIS are: *ActivityCoverage*, *ActorCoverage*, *ArtefactCove*rage. While the dynamic behaviour is considered by analysing metrics *GoalCoverage* and *PathCoverage*. They are evaluated from the technological support point of view, basically expressed in terms of number of supported activities, actors, artefacts, goal and dynamic paths. Specifically, the *ActivityCoverage* measures the percentage of process activities supported by the software systems, and it is computed as follows:

$$ActivityCoverage = \frac{\#BPAS}{\#BPA} \tag{1}$$

where, *#BPA* represents the set of business process activities, and *#BPAS* is the subset of business process activities that are supported by the software systems.

The *ActorCoverage* measures the percentage of actors involved in the business process and whose supported by the software systems. It is computed as follows:

$$ActorCoverage = \frac{\#BPActorsS}{\#BPActors} = \frac{\#(BPActors \cap SSActors)}{\#BPActors} \tag{2}$$

where, *#BPActors* is the set of business actors involved in the business process activities. *#BPActorsS* is the subset of business Actors involved in the business process and whose activities are supported by the software systems. *BPActorsS* is a subset of *BPActors* as some actors could be forced to manually execute their activities. In particular, the business actors supported by the software system are identified through the intersection of the sets of actors included in the software system design, *SSActors*, and *BPActors*.

The *ArtefactCoverage* measures the percentage of the business process artefacts implemented in the software systems. It is calculated as:

$$ArtefactsCoverage = \frac{\#BPAtfS}{\#BPAtf} = \frac{\#(BPAtf \cap SSAtf)}{\#BPAtf} \tag{3}$$

#BPAtf is the set of the business artefacts used/defined in the business process activities and modelled in the business process Model. *#BPAtfS* is the set of business artefacts that are also implemented by the software systems. *BPAtfS* is a subset of *BPAtf*, as some artefacts could not be implemented by the software systems. This subset is calculated as the intersection of the *BPAtf* set of business artefacts and the set *SSAtf* of the artefact implemented by the supporting software systems through classes modelled in the software system class

diagram.

The dynamic coverage aspects are evaluated by metrics *GoalCoverage* and *PathCoverage*. *GoalCoverage* measures the percentage of process goals supported by the software systems. It is defined as:

$$GoalCoverage = \frac{\#BPGS}{\#BPG} \quad (4)$$

where *#BPG* represents the set of the business process goals, while *#BPGS* is the subset of the business process goals that are supported for being achieved by the software systems.

The PathCoverage measures the percentage of process paths supported by the software systems. A business path indicates the sequence of business activities that have to be executed for going from an artefact status to another artefact status. Actually, it represents the achievability in the business process of an artefact status from another status of the artefact. Table 5 lists the business process paths with reference to the considered case study. The *PathCoverage* is computed as follows:

$$PathCoverage = \frac{\#BPPathS}{\#BPPath} \quad (5)$$

where *#BPPath* represents the set of the business process paths, while *#BPPathS* is the subset of the business process paths that are supported by the software systems. A business process path between two business artefact status is supported by a software system if an execution path exists that makes to evolve the system from a status equivalent to the initial business artefact status to one equivalent the final business artefacts status. Thus, *PathCoverage* is evaluated in terms of number of the business process path supported respect to the total number of process path.

Once evaluated all the coverage metrics the final value of the Technological Coverage is obtained as an average value. On the basis of the enterprise's needs, Technological Coverage can be calculated giving different weights to the metrics.

The measurement framework even measures the Technological Adequacy of the support provided by the software systems to the business process by considering both static and dynamic aspects. The technological adequacy of activities, artefacts and actors of the business process are considered from a static point of view. The Goal Adequacy is analysed for the dynamic aspect.

The *ArtefactAdequacy* is evaluated as the average of the automatic support adequacy offered to each business artefact used in the considered business process. The evaluation of this metric considers the number of business operations to be executed on the business artefacts and that are supported by the software system. It is computed as follows:

$$ArtefactsAdequacy = \frac{\sum_{\forall i \in BPAtfS} AtfA_i}{\#\text{BPAtf}} \quad (6)$$

where each component $AtfA_i$ is defined as:

$$AtfA_i = \frac{\#BMAtfS_i}{\#BMAtf_i} = \frac{\#(BMAtf_i \cap SMClass_j)}{\#BMAtf_i} \quad (7)$$

where $AtfA_i$ is the automatic support adequacy of artefact i. It is computed by considering the operations of the business artefact i that are implemented in the corresponding class of the supporting software systems. It uses the following metrics: $BMAtf_i$ is the set of business operations performed on business artefact i; $BMAtfS_i$ is the set of business operations performed on artefact i and implemented by the software system. $BMAtfS_i$ is calculated as the intersection of $BMAtf_i$ and the methods implemented in the corresponding software class $SMClass_j$

The *ActivityAdequacy* measures the adequacy of the automatic support to the execution of activity i. AA_i is computed by considering the automatic support adequacy of each business artefact j used/defined in activity i and implemented by the software systems.

$$ActivityAdequacy = \frac{\sum_{i \in BPAS} AA_i}{\#\mathrm{BPA}} \quad (8)$$

where each component AA_i is defined as:

$$AA_i = \frac{\sum_{j \in BPAtfS_i} AtfA_{ij}}{\#BPAtf_i} \quad (9)$$

where $\#BPAtfS_i$ is the set of the artefacts used/defined in activity i and implemented by the software systems.

Actor Adequacy expresses how adequate is the automatic support offered to the business actors. It is evaluated as the average of the automatic support adequacy offered to each business actor.

$$ActorAdequacy = \frac{\sum_{i \in BPActorS} ActorA_i}{\#\mathrm{BPActors}} \quad (10)$$

where each component $ActorA_i$ is defined as:

$$ActorA_i = \frac{\sum_{j \in BPAS_i} AA_{ij}}{\#\mathrm{BPA}_i} \quad (11)$$

where $ActorA_i$ is the adequacy of the automatic support offered to actor i. It is computed by considering the automatic support adequacy of each activity j the actor executes and that the software system supports. $BPAS_i$ includes all the activities actor i executes and supported by the software system.

The *GoalAdequacy* expresses how adequate is the automatic support offered to the achievement of the business goals. It is evaluated as the average of the adequacy of the automatic support for reaching each business goal. In particular, it considers all the possible paths that could occur for reaching each goal. The formula for evaluating the *GoalAdequacy* is the following:

$$GoalAdequacy = \frac{\sum_{i} GoalAdequacy_i}{\#BPGoal} \quad (12)$$

where *#BPGoal* is the number of business goals of the analysed business processes and $GoalAdequacy_i$ indicates the technological adequacy of the technological solution with reference to Goal i. $GoalAdequacy_i$ is evaluated as it follows:

$$GoalAdequacy_i = \frac{\sum_{ij} statePathAdequacy_{ij}}{\#statePath_i} \quad (13)$$

where $\#statePath_i$ is the total number of state paths making true Goal i. A path state that makes true a goal is a set of activities connecting a set of artefact states that verifies the logical expression of the goal. A goal includes more than one of such a kind of set of artefact states. $statePathAdequacy_{ij}$ is the technological adequacy of the jth state path of Goal i. It is evaluated in terms of the technological adequacy of all the activities belonging to the state path in the following manner:

$$statePathAdequacy_{ij} = \frac{\sum_{jk} ActivityAdequacy_{ijk}}{\#ActivityPath_{ij}} \quad (14)$$

where $\#ActivityPath_{ij}$ is the number of activities belonging to statePath ij and $ActivityAdequacy_{ijk}$ is the activity adequancy of one of its activities.

Once evaluated the adequacy of the automatic support offered to the business activities, actors, artefacts and goals, it is possible to calculate the final value of the Technological Adequacy, aggregating the obtained values by an average formula. Even in this case, different weights can be attributed to the metrics on the basis of business needs. The evaluation of Technological Coverage and Adequacy may not reach the highest value of 100% as there may exist some business activities that, for their nature, can just be manually executed and no automatic support can be offered to them.

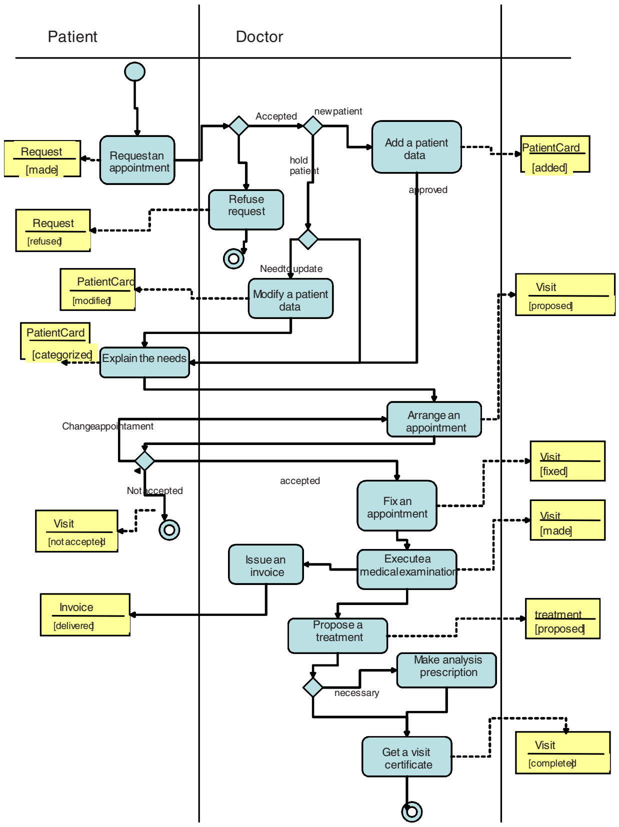

Fig. 1. Activity Diagram of the Visit Management Process

3. Case Study

To understand the applicability of the proposed enhancement of the ALBIS strategy, it was applied to a case study. The considered software system is named SMEST (Studio Medico ESTetistico - medical office aesthetic). This system was used for supporting the business activities executed within an aesthetic doctor's office, for managing patients' visits and treatments. SMEST is a software application written in C#, based on the .NET platform and using a SQLSERVER database.

The users realized that they were not adequately supported to execute some tasks and decided to require an evolution intervention. Thus, to identify the changes to perform, it was decided to evaluate the alignment degree existing between SMEST and the business processes it supported.

SMEST was analysed together with the business processes executed for handling the patients' needs, starting from the visit, to the preliminary analysis and, finally, the suggested aesthetic treatment. This analysis permitted to model the related business process through the activity diagrams depicted in Figures 1 and 2.

Figure 1 concerns the business process for managing the patients' visits, starting from their request to their completion . Figure 2 regards the execution of the medical treatment suggested after the visit. The executors of each business activity were identified in terms of actors. For example, regarding the business process of Figure 1, the Patient executes activities regarding the Request of a visit, Request of a treatment, and the Doctor executes activities such as Arrange an appointment, Execute visit, Execute treatment and so on.

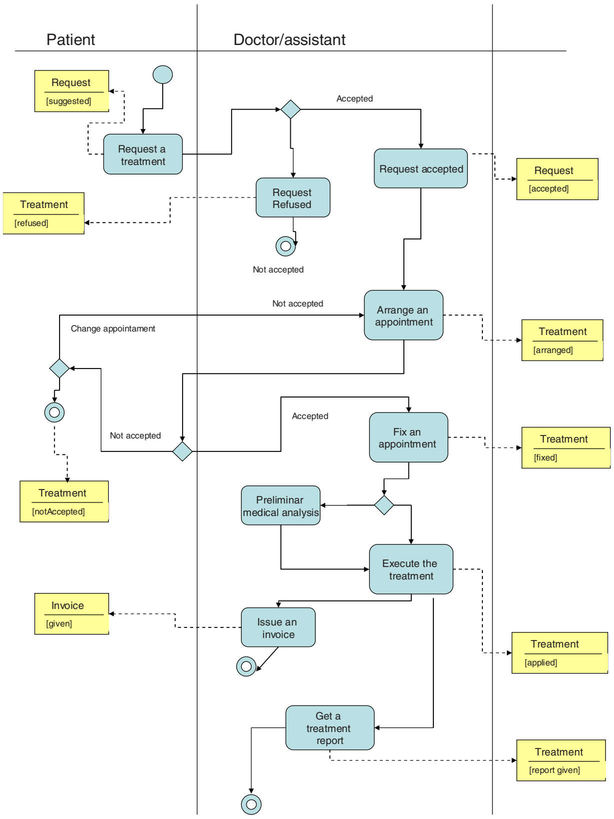

Fig. 2: Business Process Activity Diagram Treatment Management.

It is also possible to note from Figures 1 and 2 that the activity diagrams have been enriched with annotations regarding the artefacts' status. This information is relevant for helping to identify and express the business goals of the business processes and analysing their achievement by the software system.

The goals are expressed in terms of artefacts' status as Boolean expressions as shown in Table 1, where the first two goals regard the first business process and the latter two concern the second one. The goals have been identified by following the paths of the activity diagrams from the initial state of an artefact to a final or intermediate one. For example, the table shows that the first goal of the business process is Treatment visit request. It is reached by the business process execution if a visit request was made and either it was refused or accepted and, in the latter case, the Patient was registered, and the visit request was either categorized or proposed. With reference to Goal 1, it is possible to indentify three state paths whose analysis can bring to the evaluation of its technological adequacy.

Table. 1. Definition of the goal of the considered process

Goal1 - Treatment visit request	(Request.made)∩((Request.refused)∪ ((Request.accepted) ∩ ∩ ((PatientCard.added)∪(PatientCard.modified)) ∩ ∩(PatientCard.categorized)∩(Visit.proposed)))
Goal2 - Manage Visit	((Visit.notAccepted)∪((Visit.fixed)∩(Visit.made)∩ ∩ (Treatment.proposed)∩ (Visit.completed)∩(Invoice.delivered)))
Goal3 - Manage treatment request	(Request.suggested)∩ ((Treatment.refused)∪ (Request.accepted))
Goal4 - Manage treatment	(Treatment.arranged)∩((Treatment.notAccepted)∪((Treatment.fixed)∩ ∩ (Treatment.applied)∩ (Treatment.reportGiven)∩ (Invoice.given)))

The analysis of SMETS and available documentation permitted to extract a similar set of information regarding the software support of the business processes. In particular, the possible status of the artefacts managed by SMETS were identified and their achievability was identified. Figure 3 shows the artefact visit with related statuses. Once the analysis of business process and software systems was completed, the considered metrics were evaluated on the basis of their definitions. They were automatically evaluated by using a Java application, called also ALBIS [2], supporting the analysis and computing activities. ALBIS was implemented for facilitating the application of the proposed approach, and automatically support the modelling and measurement of the alignment existing between business processes and software systems.

Tables 2, 3, and, 4 include the results of the evaluation of the metrics before and after some changes were executed on the software system. Table 2 shows the analytical data measured for each activity, artefact and actor, and Table 3 includes just a summary of the evaluation of the metrics considered in the previous version of ALBIS. Table 3 shows that, if just the static aspect are considered, the aggregation of the static measures leads to Technological Coverage of 0.531, and to a value of 0.450 computed for the Technological Adequacy. By considering the new dynamic metrics, the values change and Table 4 shows that both worsen. In particular, the Technological Coverage, reaches a value of 0.519 and a value of 0.405 is achieved from the Technological Adequacy. Actually, Table 4 includes the additional values regarding the goal metrics and path metrics. Their introduction considers new business aspects that are not supported by the software system and contributes to decrease the values of both technological coverage and adequacy. The analysis of these additional aspects permitted to highlight new opportunities to improve the software system with the aim of increasing the existing alignment level. Table 5 helps to interpret these aspects. It includes some examples of state transition paths with reference to the artefacts states. Each state transition path is defined in terms of the business activities that have to be performed for passing from an initial state of an artefact to its final state. The paths have been recovered by analysing the activity diagram in Figures 1 and 2. The fourth column of the table lists the paths, while the subsequent column reports which are the covered paths before the execution of software changes. Analysing this information helps to evaluate the path state adequacy metric. The analysis of all the gathered data highlighted that the aspects not considered mainly regard the management of the visit and treatment request. For example, the path evaluation regarding activity “arrange an appointment” highlighted that some change are needed. Thus, the new metrics allowed to consider some necessary change not highlighted by the basic metric.

Thus, after the analysis of the obtained measures one of the changes that were considered for improving the software system support regarded the development of a web application to allow the patient sending an on-line request for a visit and/or a treatment.

This contributed to increase the value of the actor coverage. In particular, the value of the metric regarding the Patient actor reaches value 1.000, increasing even the actor adequacy. Other changes were executed by introducing a module report, to create medical certificate and invoice.

After the changes were executed, the technological coverage and adequacy improved as shown in the last column of Tables 2, 3 and 4.

The change execution also permitted to improve the path transition coverage, as shown in the last column of Table 5, and this implied also the increasing of the Goal Coverage.

Table 2. Technological Adequacy and Coverage measures

METRICS	VALUES BEFORE CHANGES	VALUES AFTER CHANGES
Actors Adequacy: $ActorA_i$		
$Actor_1$: *Patient*	0.000	1.000
$Actor_2$: *Doctor/Assistant*	0.500	0.889
Sum:	**0.500**	**1.889**
Artefact Adequacy : $AtfA_i$		
$Atfa_1$: *PatientCard*	1.000	1.000
$AtfA_2$: *AppointmentCalendar*	1.000	1.000
$AtfA_3$: *Treatment*	0.750	1.000
$AtfA_4$: *Visit*	0.750	1.000
$AtfA_5$: *Pathology*	1.000	1.000
$AtfA_6$: *Application Therapy*	1.000	1.000
$AtfA_7$: *ManageMedicine*	0.000	0.000
$AtfA_8$: *Invoice*	0.000	1.000
$AtfA_9$: *Request*	0.000	1.000
Sum:	**5.500**	**8.000**
Activity Adequacy: AA_i		
AA_1: *Request an appointment*	0.000	1.000
AA_2: *Refuse Request*	0.000	1.000
AA_3: *Add a patient data*	1.000	1.000
AA_4: *Explain the needs*	0.000	1.000
AA_5: *Modify a patient data*	1.000	1.000
AA_6: *Arrange an appointment*	1.000	1.000
AA_7: *Fix an appointment*	1.000	1.000
AA_8:*Execute medical examination*	0.000	0.000
AA_9: *Propose a treatment*	1.000	1.000
AA_{10}:*Make analysis prescription*	1.000	1.000
AA_{11}: *Issue an invoice*	0.000	1.000
AA_{12}: *Get a Visit certificate*	0.000	1.000
AA_{13}: *Request a treatment*	0.000	1.000
AA_{14}:*Treatment Request refused*	0.000	1.000
AA_{15}:*Treatment Request accepted*	0.000	1.000
AA_{16}: *Arrange an appointment*	1.000	1.000
AA_{17}: *Fix an appointment*	1.000	1.000
AA_{18}: *Preliminar medical analysis*	1.000	1.000
AA_{19}: *Execute the treatment*	0.000	0.000
AA_{20}: *Issue an invoice*	0.000	1.000
AA_{21}: *Get a treatment report*	0.000	1.000
Sum:	**9.000**	**19.000**
Goal Adequacy		
$GoalA_1$: *Treating visit request*	0.500	1.000
$GoalA_2$: *Manage visit*	0.200	0.900
$GoalA_3$: *Manage treatment*	0.000	1.000
$GoalA_4$:*Manage Treatment Request*	0.375	0.875
Sum:	**1.075**	**3.775**

Table 3. TA and TC values obtained for ESMEST

METRIC NAME	VALUE BEFORE CHANGES	VALUE AFTER CHANGES
Actor Coverage (ActorC)	0.500	0.950
Artefacts Coverage (AtfC)	0.667	0.889
Activity Coverage (AC)	0.428	0.904
Technological Coverage	**0.531**	**0.914**
Actor Adequacy (ActorA)	0.263	0.994
Artefacts Adequacy (AtfA)	0.612	0.889
Activity Adequacy (AA)	0.476	0.857
Technological Adequacy	**0.450**	**0.913**

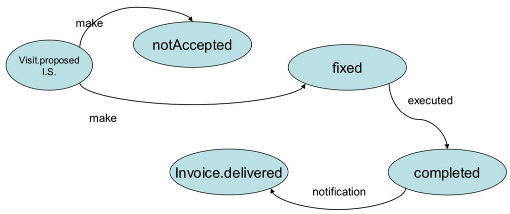

Fig. 3. State diagram of Software System SMEST.

Table 4. TA and TC values obtained for ESMEST with new metrics

METRIC NAME	VALUE BEFORE CHANGES	VALUE AFTER CHANGES
Actor Coverage (ActorC)	0.500	0.950
Artefacts Coverage (AtfC)	0.667	0.889
Activity Coverage (AC)	0.428	0.904
Goal Coverage(GoalC)	0.500	1.000
Path Coverage (PathC)	0.500	0.889
Technological Coverage	**0.519**	**0.926**
Actor Adequacy (ActorA)	0.263	0.994
Artefacts Adequacy (AtfA)	0.612	0.889
Activity Adequacy (AA)	0.476	0.857
Goal Adequacy (GoalA)	0.269	0.944
Technological Adequacy	**0.405**	**0.921**

Table 5. A fragment of the state transition coverage before/after change.

	Start state	Final state	Path	Coverage Before change	Coverage After change
1	Request.made	Request.refused	<Request an appointment, Refuse request>	no	yes
2	Request.made	Request.accepted	<Request an appointment>	no	yes
4	PatientCard.added	Visit.proposed	<Explain the needs, Arrange an appointment>	yes	yes
5	PatientCard.modified	Visit.proposed	< Modify a patient data, Explain the needs, Arrange an appointment >	yes	yes
6	Visit.proposed	Visit.NotAccepted	<Arrange an appointment >	yes	yes
7	Visit.proposed	Visit.fixed	<Arrange an appointment, Fix an appointment >	yes	yes
8	Visit.fixed	Visit.made	<Execute a medical examination>	yes	yes
9	Visit.fixed	Treatment.proposed	< Execute a medical examination, Propose a treatment>	yes	yes
10	Visit.fixed	Invoice.delivered	< Execute a medical examination, Issue an invoice>	no	yes
11	Visit.fixed	Visit.completed	< Execute a medical examination, Propose a treatment, make analysis prescription>	no	yes

4 Conclusions

This paper deals with the problem of managing the alignment level existing between a business process and the supporting software systems. This issue was widely recognized as relevant for the business process performance. In the paper, the attention has been focalized on the business goals definition and their impact on the alignment of business process and software system. Thus, it proposes an approach consisting in a set of metrics helping in measuring the alignment level.

The framework involves the evaluation of a set of metrics for evaluating the Technological Coverage and Technological Adequacy of the software systems used for supporting a business process. Its application involves the analysis of all the goals, activities, artefacts, actors and operations, of the business process and software components. The dynamic behaviour of the software system with reference to the business goal is considered.

The results of the evaluation of the metrics allow for the identification of a possible misalignment. In particular, they provide a measure of the alignment degree. If misalignment emerges, the achieved alignment values, together with the comparison of the business and software models and related tracing identified by performing the semantic analysis, can be used for indicating which software activities, classes or methods are missing for achieving the target alignment. Then, evolution activities can be easily planned and executed for improving the alignment level and guaranteeing the most efficient and effective execution of the business process.

References

[1] Aversano, L., Grasso, C., Tortorella, M., 2010a. "Measuring the Alignment between Business Processes and Software Systems: a Case Study", ACM Symposium on Applied Computing, Sierre, Switzerland, pp. 2330-2336.

[2] Aversano, L., Marulli, F., Tortorella, M., 2010b. "Recovering Traceability Links between Business Activities and Software Components". Conference on ENTERprise Information Systems, CENTERIS 2010, Advances in Software Engineering, Communications in Computer and Information Science. Springer-Verlag, Vol. 109, pp.385-394.

[3] Aversano, L., Grasso, C., Tortorella, M., 2012. "A Literature Review of Business/IT Alignment Strategies". Procedia Technology. Vol. 5, Elsevier Ltd, Vilamoura, Portugal, pp. 462-474.

[4] Basili, V.R., Caldeira, G., Rombach, H.D., 1994. The goal question metric approach, in "*Encyclopedia of Software Engineering*", Wiley.

[5] Becker, A.L., Prikladnicki, R., Audy, J.L.N., 2008. "Strategic Alignment of Software Process Improvement Programs Using QFD", International Conference on Software Engineering, 1st international workshop on Business impact of process improvements, Leipzig, Germany, pp. 9-14.

[6] Bubenko, J., Persson, A., Stirna, J. EKD User Guide 2001, http://www.dsv.su.se/~js/ekd_user_guide.html

[7] Byrd, T.A. Lewis, B. R., Bryan, R.W., 2006. The leveraging influence of strategic alignment on IT investment: an empirical examination, J. of Information Management, Elsevier 43(3), pp 308-321.

[8] Card, S., Moran, T.P., and Newell, A., 1983. The Psychology of Human-Computer Interaction, Lawrence Erlbaum Associates, Hillsdale, New Jersey, USA.

[9] Ellis, C. A., Wainer, J., 1994. Goal-Based Models of Collaboration, Collaborative Computing 1(1), pp.61-86.

[10] Gordijn, J., Yu, E., Raadt, B.V.D., 2006. E-Service Design Using I* and e3 value Modeling, IEEE Software 23(3), pp. 26-33.

[11] Henderson, J.C., Venkatraman, N., 1993. Strategic Alignment: Leveraging Information Technology for Transforming Organizations IBM Systems Journal. 32(1), pp. 4-16.

[12] Reich, B.H., and Benbasat, I, 1996. Measuring the Linkage between Business and Information Technology Objectives, MIS Quarterly 20(1), pp.55-81.