MODELING AND VERIFICATION OF ERP FUNCTIONAL REQUIREMENTS BASED ON COLORED PETRI NET

Nafisa Osman and Abd-Elkader Sahraoui

Department of computer science, Sudan University of science and technology, Khartoum, Sudan LAAS-CNRS, avenue du Colonel Roche, F-31077 Toulouse, France Université de Toulouse; F-31077 Toulouse, France

ABSTRACT

The rise of enterprise resource planning (ERP) systems has been a major event in the software industry and it became a solution for most enterprises to manage their data and business processes. Successful ERP implementations can reduce costs by improving efficiency then lead to improved company performance and better competitive edge. Despite these benefits and the age of ERP existing for several decades still high percentage of implementation failures are documented. ERP is packaged software designed by following the best practice from the specific industry to support typical business processes in the entire industrial field, it was designed by ERP vendors and used by the organization which implement it. Since the designer and user are two independent entity misalignments between user's needs and the software design are often happen. The misalignment define new specific requirements must be embedded into selected ERP.

Requirement engineering (RE) is a main part and initial activity of software engineering concern about defines stakeholder requirements, needs and desire. Requirements engineering is the basis for efficient software implementation and quality management. Tools and theories which support RE in general are numerous nowadays; however, the task of providing a tools and theories that specializes in Requirements engineering for Enterprise resource planning systems has been addressed rarely. For that; this paper discusses modelling and verification of ERP functional requirements based on colored Petri nets (CPN) after evaluation of different Business process modelling techniques by using analytical hierarchy process (AHP). CPN considered one of powerful business process modelling techniques and using it help in stakeholder involvement and appropriate organization's business process representation. The nature of colored Petri nets that help in verification of internal completeness and consistency of ERP functional requirements model.

KEYWORDS

Requirements engineering, enterprise resource planning, requirements modelling, colored Petri nets, requirements verification, business process modelling.

1. INTRODUCTION

Today organizational processes and supply chains become more complex. They involve complex business processes and functions involving different departments, with need for up to date information and critical linking with upstream and downstream partners. To fulfill these business objectives many business information technologies emerged during last decades. In the last three decades, organizations have moved from standalone business information system applications to integrated and flexible enterprise information systems. The rise of enterprise resource planning

DOI: 10.5121/ijsea.2021.12303

(ERP) systems has been a major event in the software industry and it became a solution for most enterprises to manage their data and business processes(F. Chen et al., 2013).

Successful ERP implementations can reduce costs by improving efficiency through computerization; and enhance decision-making by providing accurate and updated organizationwide information: then lead to improved company performance and better competitive edge(Botta-genoulaz et al., 2018; Haddara & Elragal, 2013). Despite these benefits and the age of ERP existing for several decades still high percentage of implementation failures are documented(Haddara & Elragal, 2013). A recent survey published in panorama consulting annual report at 2016 summarizes that 46% organization receive 50 % or less of the expected benefits from their ERP implementation(Panorama Consulting Solutions, 2016), other studies state that 90% of ERP project exceed either budget or schedule and delivered only 30 % of committed goals(Shaul & Tauber, 2013; Zare & Ravasan, 2014). These statistics imply that the ERP projects are one of the most difficult system development projects. Unlike the other information system, the major problems of ERP implementations are not technologically related issues such as complexity, comparability ... etc. but most about organization and human related issues such as resistance to change, organizational culture, incompatible business process, project mismanagement, etc(Helo et al., 2008). The development of custom ERP applications is expensive; therefore, ERP systems are mostly selected as off-the-shelf software packages(Klaus et al., 2000). ERP is packaged software designed by following the best practice from the specific industry to support typical business processes in the entire industrial field, it was designed by ERP developers / vendors and used by the organization which implement it. Since the designer and user are two independent entity misalignments between user's needs and the software design are often happen(Kanchymalay et al., 2013). This misalignment creates negative impact on the implementing organization and implementations are often unsuccessful(van Beijsterveld & van Groenendaal, 2016). This unsuccessful implementation affects not only on the implementing company, but also the ERP vendors because they have to pay compensation to the implementing firm.

Resolving misfits can be achieved by either customizing the ERP system or by adopting the business processes of the organization to fit the requirements of the ERP system(van Beijsterveld & van Groenendaal, 2016). The misalignment define new specific requirements must be embedded into selected ERP, Requirement engineering is a main part and initial activity of software engineering concern about defines stakeholder requirements, needs and desire(ur Rehman et al., 2013). Requirements engineering for system based on off-the-shelf software component has been regarded an important sub area of requirements engineering. In this type of systems, requirements engineering takes place at the early phase related to ERP package selection(Asghar & Umar, 2010; Daneva & Wieringa, 2010). The requirement for ERP concern about the business processes and the data flow that the ERP should support it. These requirements reflect the organizational units in one or more companies for ERP that helps to solve coordination and collaboration problem related to processing. Requirements engineering (RE) for ERP is about composition and reconciliation conflicting demands. Usually start with general set of business processes and data requirements. Then help to explore standard ERP package functionality to see how closely it matches the ERP adopting organization's process and data needs(Daneva, 2004; Daneva & Wieringa, 2010). The paper concern about modeling and verification of functional ERP requirements based on business process modeling, especially Models play an important role in the development process of an information system(Giaglis, 2014; Wil M P van der Aalst & Stahl, 2011). They can describe information system to be designed and business process supported by it in certain form (e.g. textual or graphical) (Wil M P van der Aalst & Stahl, 2011). We can define business process as collection of related structured and logically ordered set of activities or tasks that produce a specific service or product (serve particular goal) for particular customer(Brusakova & Kossukhina, 2011; Röder et al., 2015).

There are several well-known business process modeling techniques within each approach but, according to the nature of enterprise resource planning system as a kind of embedded workflow management system (WFMS) (Tealab & Galal-Edeen, 2007), in ERP and WFMS both classes of systems focus on business process, but the approach taken by them are different. We select the most popular workflow oriented approach techniques are (system flowchart, event driven process chain, business process modelling notations, and colored Petri net). In the next section we will give a brief overview about these different modelling techniques.

2. MODELLING TECHNIQUES

2.1. System flowcharts

System flowcharts are one of the oldest forms of diagrams used in information systems(Dunn, 2006; Giaglis, 2014). While they were originally used to document computer programs, they are also described as a means of documenting information flows and related business processes (Dunn, 2006). They also represent nonetheless the most basic type of diagram for communicating business process flows. Flow charts are used predominantly in software engineering, but their simplicity and ease of use have enabled managers and business owners to adopt this technique for organizational purposes as well (Aldin & Cesare, 2009).

System flowcharts capture inputs and outputs, information activities, storage, and entities. There are a large number of variants and extensions to the basic symbols, including ANSI and ISO notations, and many variants on how the symbols are combined to model a particular process(Dunn, 2006).

Strength of system flowcharts is that they do capture flow of control via decision points, and can show manual versus automated operations and inputs. However, they does not support a breakdown of the activities (Dunn, 2006)

2.2. Event driven process chain diagrams

EPC diagrams are a component of the Architecture of Integrated Information Systems (ARIS) framework developed by Scheer (1992), described as subsequently adapted for use in SAP AG's R/3 reference model. EPC diagrams are used in ARIS and SAP R/3 to describe business processes, and in ARIS to represent the "control view" which links functions, organizations, and data. Basic EPCs as consisting of activities, events, and connectors indicating flow of control between activities and events. The intent underlying an EPC is to model the complete chain of activities that occur in response to a business event, and thus describe a business process. Events trigger activities as well as describe the outcome after an activity has occurred, e.g. the receipt of a customer order, or the acceptance of a customer order.(Dunn, 2006)

A key strength of EPC diagrams is that they are easy to understand (Dunn, 2006), Unfortunately, neither the syntax nor the semantics of an EPC are well defined. As a result, an EPC may be ambiguous. Moreover, it is not possible to check the model for consistency and completeness. These problems are serious, because EPCs are used as the specification of business processes and need to be processed by the ERP and WFM systems. The absence of formal semantics also hinders the exchange of models between tools of different vendors and prevents the use of powerful analytical techniques (Wil M P der Aalst, 1999). However, identification of the appropriate events to be modeled (e.g. those which trigger activities) can also be challenging. In addition, EPC diagrams also do not have an explicit construct to model controls, although the activity construct may be used to represent controls such as authorization or verification, and

examination of the organizational units involved may provide some sense of segregation of duties. There is no representation for the resources required for a particular activity beyond that of the organizational unit performing the activity (Dunn, 2006).

2.3. Business Process Modelling and Notation

Business Process Model and Notation (BPMN) is one of the popular standardized notations for software process modelling (Dechsupa et al., 2018) that provides graphical notation for specifying business processes based on traditional flowcharting techniques (Wang et al., 2006). The notation inherits and combines elements from a number of previously proposed notations for business process modeling, including the XML Process Definition Language (XPDL) and the Activity Diagrams component of the Unified Modelling Notation (UML)(Remco, 2008) . A BPMN design model illustrates the overall control flow and data flow within a software process model (Dechsupa et al., 2018)

BPMN was invented by the Business Process Management Initiative (BPMI) and was maintained by the Object Management Group in 2005. It is a graphical standard notation used for describing a business procedural logic, business process or work flow. In software development, the BPMN is a design model that bridges the communication gaps among developers(Dechsupa et al., 2018). The main strength of BPMN it's easy to understand the flow of the process (Search et al., n.d.)but with over of 100 symbols(Steps, 2013) there are ambiguity and confusion in sharing of BPMN (Search et al., n.d.)also the lack of semantic (Dijkman et al., n.d.). Both of these are considered main weakness.

2.4. Colored Petri net

Petri nets are a graphical and mathematical modelling tool applicable to many systems .(Sackmann et al., 2006) It combines "visual representation using standard notation with an underlying mathematical representation" (Aldin & Cesare, 2009). Petri nets is a promising tool for describing and studying information processing systems that are characterized as being concurrent, asynchronous, distributed, parallel, nondeterministic, and/or stochastic.(Sackmann et al., 2006) because of their characteristics: simplicity, representation power comprising concurrency, synchronization and resource sharing, strong ability of their mathematical analysis and application of software tools(Bosilj-vuksic, 2000)

As a graphical tool, Petri nets can be used as a visual-communication aid similar to flow charts, block diagrams, and networks. As a mathematical tool, it is possible to set up state equations, algebraic equations, and other mathematical models governing the behavior of systems.(Sackmann et al., 2006)

The concept of the Petri net has its origin in Carl Adam Petri's dissertation submitted in 1962 to the faculty of Mathematics and Physics at the Technical University of Darmstadt, West Germany. (Sackmann et al., 2006) Nowadays, Petri nets are used for modelling computer software, hardware, control flow and business processes (Aldin & Cesare, 2009)



Figure 1: Petri net component Source :(Fortier & Michel, 2003)

Petri net contains four components. These four Petri net modelling components are place, transition, arc, and token. Places are represented graphically as a circle, transitions as a bar, arcs as directed line segments, and tokens as dots (check Figure [1]). Places are used to represent possible system components and their state.. Transitions are used to describe events that may result in different system states.. Arcs represent the relationships that exist between the transitions and places. You can think of the arc as providing a path for the activation of a transition. Finally, tokens are used to define the state of the Petri net. Tokens in the basic Petri net model are no descriptive markers, which are stored in places and are used in defining Petri net marking (Fortier & Michel, 2003).The marking of Petri net determined by distribution of token over places of the net Transitions can change the marking through firing and to fire a transition, it must be enabled. A transition is enabled if there is at least one token in each of its input places when transition firing, one token is removed from each of its input place and one new token added to each of its output place. (Wil M P van der Aalst & Stahl, 2011)

Petri nets can model the behaviour of a system, but three serious weakness are:(Wil M P van der Aalst & Stahl, 2011)

- 1. Large network size.
- 2. Limited expressive power.
- 3. No explicit modelling of time.

Colored Petri net is the first extending to provide extra functionality it from indistinguishable to distinguishable token by adding value referred as color. (Wil M P van der Aalst & Stahl, 2011) The basic differences between Petri net and CPN, Petri net marking described by the number of tokens for each place. to describe the marking of colored Petri net we also need to describe the value of these tokens (Wil M P van der Aalst & Stahl, 2011)

Colored Petri net is a Petri net in which every place has a type, and every token has a value (i.e., color) that complies with the place type. An arc in colored Petri net may have arc inscription. An arc inscription is an expression with some variable that evaluate to a multi sets. A transition has guard. a guard is a Boolean expression. The next section presents the selection process of one modeling technique to model the functional requirements

3. COMPARISON OF BUSINESS PROCESS MODELLING TECHNIQUES

The previous section introduces the four business process modelling techniques; this section provides comprehensive selection of business process modelling technique by using analytical hierarchy process (AHP) approach. The AHP approach developed by Satty (1980) and has been applied to a wide variety of decisions and the human judgment process(C.-F. Chen, 2006). It was

developed to help managers make more effective decisions by structuring and evaluating the relative attractiveness of competing options or alternatives (Handfield et al., 2002).

The approach is used to construct an evaluation model and has criterion weights. It integrates different measures into a single overall score for ranking decision alternatives. Applying it usually results in simplifying a multiple criterion problem by decomposing it into a multilevel hierarchical structure(C.-F. Chen, 2006).

Applying the AHP procedure involves three basic steps: (1) decomposition, or the hierarchy construction; (2) comparative judgments, or defining and executing data collection to obtain pair wise comparison data on elements of the hierarchical structure; and (3) synthesis of priorities, or constructing an overall priority rating(C.-F. Chen, 2006)

Step 1: hierarchy construction

In order of selection BPM techniques we define level of criteria includes (simplicity, understandability, expressiveness, formality and simulation). Once a hierarchy has been developed, one moves to data collection, thus having the pair wise comparisons needed to determine the relative importance of the elements in each level. The decision makers begin the prioritization procedure to determine the relative importance of the elements in each level(C.-F. Chen, 2006).

Step 2: Construct a pair wise comparison

In using AHP, the decision maker must specify his judgments of the relative importance of each criterion's contribution towards achieving the overall goal. The evaluations would then be solicited using questions such as "What is the importance of cost reduction relative to an increase in reliability in accomplishing our overall goal?" Similar pair-wise comparisons for other criteria can be done to generate Saaty's pair-wise comparison matrix. To answer these questions must consider 9-point intensity of relative importance scale (check figure [2]). The consistency ratio is the ratio of the decision maker's inconsistencies and inconsistencies resulting from randomly generated preferences (Partovi et al., 1990).

Intensity of Relative Importance	Definition	Explanation
1	Equal importance	Two activities contribute equally to objective 1.
3	Moderate importance of one over another	Experience and judgment slightly favor one activity over another.
5	Essential or strong importance	Experience and judgment strongly favor one activity over another.
7	Demonstrated importance	An activity is strongly favored, and its dominance is demonstrated in practice.
9	Extreme importance	The evidence favoring one activity over another is of the highest possible order of affirmation.
2, 4, 6, 8	Intermediate values between the two adjacent judgments	When a compromise is needed.
Reciprocals of the above nonzero numbers	Reciprocal for inverse comparison	

Figure 2: 9-point intensity of relative importance scale Source: (C.-F. Chen, 2006)

Compare and construct a pair wise comparison of distinct criteria based on the objective as presented in Table [1]

Criteria	simplicity	Understand-ability	Formality	expressiveness	Simulation
Simplicity	1	1	0.33	0.33	1
Understand-ability	1	1	0.33	0.33	1
Formality	3	3	1	3	1
Expressiveness	3	3	0.33	1	3
Simulation	1	1	1	0.33	1

Table 1: criteria of selection

Step 3: calculating normalization array and weight

After all pair wise comparison matrices are formed, first, the pair wise comparison matrix, is normalized by equation (1), and then the weights are computed by equation (2) (C.-F. Chen, 2006)

$$Aij = \frac{aij}{\sum_{i=1}^{n} aij}$$
For all j=1,2,3 ...,n.
$$Wi = \frac{\sum_{j=1}^{n} Aij}{n}$$
For all i=1,2,3,...,m.
(1)
(2)

The normalized array and weight presented in table [2]

Simplicity	Understand-ability	Formality	Expressiveness	simulation	Weight
0.111111	0.111111	0.110368	0.066132	0.142857	0.108316
0.111111	0.111111	0.110368	0.066132	0.142857	0.108316
0.333333	0.333333	0.334448	0.601202	0.142857	0.349035
0.333333	0.333333	0.110368	0.200401	0.428571	0.281201
0.111111	0.111111	0.334448	0.066132	0.142857	0.153132

Step 4: Check consistency ratios

The value computed from the Consistency Index (C.I.) is then compared with Random Index (R.I) values. Satty & Kearns in a previous publication presented consistency check R.I values. C.R. = C.I. /R.I. = 9.106222 < 10% the comparison is consistent.

Step 5: Determine weights of BPM techniques alternatives based on distinct criterion

The same process as presented in steps [2, 3] is repeated to determine the measurable scale and criteria weights of BPM techniques alternatives based on a distinct criterion. The researchers calculated criteria weights for simplicity case presented in Table [3] criteria weights for BPM techniques related to rest of criteria are obtained following the same process.

Simplicity	Flow	EPC	BPMN	CPN
	cnart			
Flow chart	1	3	3	0.33
EPC	0.33	1	3	0.25
BPMN	0.33	0.33	1	0.25
CPN	3	4	4	1

Table 3: simplicity pair wise comparison for BPM techniques alternative

Step 5 : Determine overall weights

Computed criteria weights from distinct normalized matrix (step 4) is collected and multiplied by the original criteria weights (step 2). Results are presented in Table [4]. From the weights presented we can figure the CPN its high weight that means it will be selected for modelling ERP business requirements

Weight
.181
.144
.23
.43

Table 4: weights of different BPM techniques

This step it's a heart of methodology which using business process modeling, which is concern with conducting behavioural model, we have to identify the internal behavior and data entities which are represent modelling functional requirement.CPN is a business processes modeling technique used to modelling complex concurrent system with different features can present business processes details for business orientation system like ERP.

Depending on the details presented above, CPN can model the data related to business processes by using color and modelling business process constrains and rules by using guard and arcs inscription. Transitions present event or actions

4. VERIFYING CPN

Software requirement specification should be comprehensive statement intended behavior of software (Heimdahl & Leveson, 1996), requirements should be completed, consistent and correct (3cs) especially Errors in requirements are pervasive, dangerous, and costly. One promising approach to reducing requirements errors is to apply formal methods during the requirements phase of software development. By a formal method, we mean a development method based on some formalism, such as a formal specification notation or a formal analysis technique. A formal requirements specification can reduce errors by reducing ambiguity and imprecision and by making some instances of inconsistency and incompleteness obvious. Formal analysis can detect many classes of errors in requirements specifications, some of them automatically(Heitmeyer et al., 1996)

CPN hold business processes details which represents functional requirement so by using formal verification of model we already verified the requirements within it. By using CPN tool to produce state space model and simulation to check specific properties of CPN mapping into

requirement properties. But firstly, we will define the 3cs then we will present the details related to CPN analysis to conclude the mapping between properties and the 3cs.

4.1. concept of 3cs

• Completeness

Completeness is considered to be the most difficult of the specification attributes to define and incompleteness of specification the most difficult violation to detect the requirements document must exhibit three fundamental characteristics: (1) No information is left unstated or \to be determined, (2) The information does not contain any undefined objects or entities (3) No information is missing from this document. The first two properties imply a closure of the existing information and are typically referred to as internal completeness. The third property, however, concerns the external completeness of the document. External completeness ensures that all of the information required for problem definition is found within the specification. This definition for external completeness clearly demonstrates why it is impossible to define and measure absolute completeness of specification.(Zowghi & Gervasi, 2002)

• Consistency

Consistency requires that no two or more requirements in a specification contradict each other. Or refers to situations where a specification contains no internal contradiction (Zowghi & Gervasi, 2002). The consistency mean the specification free from conflicts and undesired non-deterministic (Heimdahl & Leveson, 1996).

• Correctness

From a formal point of view, correctness is usually meant to be the combination of consistency and completeness. From a practical point of view, however, correctness can be more pragmatically defined a satisfaction of certain business goals. This indeed is the kind of correctness which is more relevant to the customer, whose goal in having a new system developed is to meet his overall business needs(Zowghi & Gervasi, 2002)

4.2. Petri Net Analysis

The objective of using Petri nets in system study is to draw important conclusions about the system without going for time and cost ineffective trial and error prototyping. To do so, the first step is to model the system. Once a model is ready, the next task is to analyze the model to draw conclusions about the properties of the model and hence about the actual system. Then only one can answer questions like what the system behaviour is supposed to be under specific operational conditions, what properties are inherent to the structure of the net, what to expect and what not to expect from the system during operation and whether there is any pitfall in the system design which must be avoided in operational phase(Halder & Venkateswarlu, 2006; Murata, 1989). There are three major approaches for PN analysis :(1) Behavioural approach, which is a tree-based approach, (2) Structural approach, which is a matrix-based approach and (3) Reduction or Refinement approach, which is a net simplification approach. (Halder & Venkateswarlu, 2006). We will focus here on behavioural approach.

• Behavioural approach

This approach deals with the behavioural properties of PN. Behavioural properties are those which are dependent on the initial marking. It's called reach-ability analysis also, Reach-ability analysis is done by construction of reach-ability tree and is also called a cover ability tree. A reach-ability graph consists the set of markings reachable from m0, this set represented into graph consist of nodes and edges. Each node corresponds to reach-ability marking of the net, each edge to transition moving the net from one marking to another one (W. der Aalst & Stahl, 2011)

• Reach-ability

Given a Petri net, given an initial marking m0, given another marking mr; the question is whether there exists a sequential firing of transitions which will bring the net from m0 to mr. If the answer is 'yes', then mr is said to be Reachable from m0. The set of all possible markings reachable from m0, is called the reach-ability Set, denoted by the symbol R(m0) for a given PN. Reach-ability set is defined for a given PN, for a given initial marking m0. This dependency on initial marking clearly reveals that reach-ability is a behavioural property.(Halder & Venkateswarlu, 2006)



Figure 3: (a) petri net graph (b) reachability tree for (a)

Boundedness

A PN is called k-bounded with respect to an initial marking m0, if each place in the net gets at most k tokens for all markings belonging to the reach-ability set R (m0), where k is a finite positive integer.(Halder & Venkateswarlu, 2006; Wil M P van der Aalst & Stahl, 2011)

Mathematically, for Bounded-ness it should always happen that with respect to an initial Marking m0, $m(pi) \le k \forall i \in [1, n]$ and this should happen $\forall m \in R(m0)$

If k = 1, then the PN is called Safe. Therefore, safeness is a special case of bounded-ness. In a safe PN, a place can either contain no token or it can contain only one token. In a safe net, there exists no such marking belonging to the reach-ability set (with respect to an initial marking) for which number of tokens in any place of the PN exceeds one.(Halder & Venkateswarlu, 2006)

• Liveness

A PN is called Live with respect to an initial marking, if for every marking belonging to the reach-ability set; it is possible to fire all the transitions at least once by some firing sequence. Mathematically, a PN is called Live with respect to an initial marking m0, if $\forall m \in R(m0)$, it is possible to fire all the transitions at least once by some firing sequence.(Halder & Venkateswarlu, 2006)

Dead Transition

Transition t of a Petri net is dead if and only if t is not enabled at any reachable marking.

4.3. Mapping Petri Net Properties to Check Requirement Completeness and Consistency

• Requirement completeness verification

The CPN is complete if

• The CPN is reachable

According to the above, Reach-ability includes checking enabling and firing rules , by recalling firing rules transition enabling when each one of its input places is marked with at least token , firing enabled transition perform execution , the execution of Petri net cause it marking to change by moving token from input place and depositing into output place . Depend on above we check input, output and event completeness, as information stated and defined.

• The CPN Strongly connected

A Petri net is strongly connected iff, for every pair of nodes (i.e., places and transitions) x and y, there is a path leading from x to y(W M P Van Der Aalst, 2003)

• The TCPN is bounded-ness, bounded express the required token and not more than required its available at each reachable marking

• Requirement consistency verification

The CPN is consistent if

- The CPN is deadlock free or live, deadlock freedom reflects no contradiction or stop point within TCPN.
- CPN is Dead transition free,

By using CPN tool to generate state space report based on reach ability tree and developed java program, we checked specific CPN properties after considering these properties can help to verify requirement internal completeness and consistency.

5. **DEMONSTRATION**

Company [A] seeking to implement ERP package. The company choose Odoo package for different reasons such as open source, technical and functional support availability, etc. we implement the proposed methodology steps as following.

Based on requirement elicitation and documentation phase we get different documents related to each business department in company A included in ERP implementation project. We choose purchase request business process to demonstrate our methodology.

5.1. Build CPN

Corresponding to the purchase department requirement we build CPN model as follow:



Figure 4: purchase request business process details

- opdons
vpurchase requirement.cpn
Step: 0
Time: 0
Options
► History
▼Declarations
Standard priorities
Standard declarations
colset UNIT
► colset BOOL
colset INT
► colset INTINF
colset TIME
colset REAL
▼colset STRING = string;
▼colset name=STRING;
▼colset order_date =STRING;
▼colset purpose = STRING;
colset department = STRING;
colset department_code=STRING;
▼colset requested_person = STRING;
colset product_name=STRING;
▼colset currency = STRING;
▼colset notes= STRING;
<pre></pre>
colset quantity = INT;
▼colset cost = INT;

Figure 5: purchase request form data

Figure [4] present the procedures to create and get final approval for purchase request in company [A], which include 8 states; starting with draft as beginning of the process then each request must be confirmed from corresponding department head. the purchase request need to checked from material planning if the request approved we can get material planning confirmation. The business process flow contains purchase department reviewing then recommendation the process ends by supply chain management approval.

Figure [5] represents the data requested by customer which include in purchase request for in company [A]. And the combination of these variables represents the data record assigned to each state in the model.

5.2. Verifying CPN

As we discussed the CPN model built in previous step and shown in figure [4] represents the customer requirements. During this step we generate state space for CPN model [check figure [6]]then we use Java program to verify the CPN model based on CPN and corresponding state space then present the details in understandable form.

		g
ort generated. Sat Dec	5 03:59:35	2020
atistics		
tate Space		
Nodes: 7		
Arcs: 6		
Secs: 0		
Status: Full		
cc Graph		
Nodes: 7		
Arcs: 6		
Secs: 0		
undedness Properties		
est Integer Bounds	Upper	Lower
undedness Properties est Integer Bounds New Page approved 1 New Page of im 1	Upper	Lower 0
undedness Properties est Integer Bounds New_Page.approved 1 New_Page.cofirm 1 New_Page.draft.1	Upper 1 1	Lower 0 0
est Integer Bounds New_Page'approved 1 New_Page'cofirm 1 New_Page'draft 1 New_Page'draft 1 New_Page'final approva	Upper 1 1 1	Lower 0 0 0
undedness Properties est Integer Bounds New_Page.approved 1 New_Page.cofirm 1 New_Page.draft 1 New_Page.final_approva	Upper 1 1 1 1	Lower 0 0 0
undedness Properties est Integer Bounds New_Page'approved 1 New_Page'cofirm 1 New_Page'draft 1 New_Page'final_approva New Page'final_approva	Upper 1 1 1 1 1	Lower 0 0 0 0
undedness Properties est Integer Bounds New_Page approved 1 New_Page cofirm 1 New_Page draft 1 New_Page final_approva New_Page recommended 1 New_Page rejection 1	Upper 1 1 1 1 1 1	Lower 0 0 0 0
windedness Properties est Integer Bounds New_Page'approved 1 New_Page'draft 1 New_Page'frail approva New_Page'recomended 1 New_Page'rejection 1 New_Page'rejection 1 New_Page'rejection 1	Upper 1 1 1 1 1 1 1 1 1	Lower 0 0 0 0 0
undedness Properties est Integer Bounds New_Page approved 1 New_Page cofirm 1 New_Page final_approva New_Page recomended 1 New_Page rejection 1 New_Page reviwed 1 New_Page second confirm	Upper 1 1 1 1 1 1 1 1 1 1 1 1 1 1	Lower 0 0 0 0 0 0
est Integer Bounds New_Page'approved 1 New_Page'cofirm 1 New_Page'draft 1 New_Page'recomended 1 New_Page'recomended 1 New_Page'recomended 1 New_Page'second confirm	Upper 1 1 1 1 1 1 1 1 1 1 1 1 1	Lower 0 0 0 0 0 0 0 0
undedness Properties est Integer Bounds New_Page approved 1 New_Page cofirm 1 New_Page draft 1 New_Page final_approva New_Page recomended 1 New_Page rejection 1 New_Page reviwed 1 New_Page second_confirm	Upper 1 1 1 1 1 1 1 1 1 1 1 1	Lower 0 0 0 0 0 0 0

Figure 6: purchase request CPN model state space report

Then the state space file and CPN file being input for Java program file to define if any missing details to express about requirement internal completeness. In this case the requirements related to this request is complete according to discussion on the above, as showing in figure [7]



Figure 7: verification result

5.3. Incomplete case

The figure bellows present purchase requirements with adding new transition and missing data between SCM approval transition and final approval to see what verification result. By using the developed java program and according to scenario with state space the verification results shows unreachable marking with defining that there is incomplete data in arc between SCM approval transition and final approval state. Also there is isolated transition (passed) this means the functional requirement of purchase request process incomplete.



Figure 8: incomplete purchase request requirements



Figure 9: state space incomplete purchase request requirements

ument Selector	4 ×test_cpn.java ×
_cpn.java	<pre>import java.io.*; import java.util.*; import java.util.</pre>
	C:\WINDOWS\system32\cmd.exe
	unreachable marking / place
	not comleted data For Arc between Place Trans
	SCM approval final approval Isolated Transition
	======================================

Figure 10: verification result

6. CONCLUSION

Requirements engineering is the basis for efficient software implementation and quality management. Tools which support RE in general are numerous nowadays; however, the task of providing a tool that specializes in RE for dynamic, customizable service-centric systems (like ERP systems) has been addressed rarely. According to this the paper proposed requirement modelling and verification of ERP functional requirements based on Colored Petri nets. Colored Petri nets are a graphical and mathematical modelling tool applicable to model enterprise information system and the business process they support. Paper validated by using case study and achieved good results. Firstly, using colored Petri nets for modelling ERP functional requirement, organization's business process representation.

Secondly, the nature of colored Petri nets that help in verification of internal completeness and consistency of ERP functional requirements model.

References

- 1. Aalst, W M P Van Der. (2003). Challenges in business process management: Verification of business processes using Petri nets. Bulletin of the EATCS 80, 174-199.
- 2. Aldin, L., & Cesare, S. De. (2009). A COMPARATIVE ANALYSIS OF BUSINESS PROCESS MODELLING TECHNIQUES. Ukais, 1–17.
- 3. Asghar, S., & Umar, M. (2010). Requirement engineering challenges in development of software applications and selection of customer-off-the-shelf (COTS) components. International Journal of Software Engineering, 1(1), 32–50.
- Bosilj-vuksic, V. (2000). IDEF Diagrams and Petri Nets for Business Process Modeling : Suitability , Efficacy , and Complementary Use IDEF Diagrams and Petri Nets for Business Process Modeling : Suitability , Efficacy , and Complementary Use. January. https://doi.org/10.1007/978-94-017-1427-3
- 5. Botta-genoulaz, V., Millet, P., Botta-genoulaz, V., & An, P. M. (2018). An investigation into the use of ERP systems in the service sector To cite this version : HAL Id : hal-00266991 An investigation into the use of ERP systems in the service sector. https://doi.org/10.1016/j.ijpe.2004.12.015
- 6. Brusakova, I., & Kossukhina, M. (2011). Review of approaches in modeling adaptive business process. Proceeding of the 10th Conference of Open Innovations Association FRUCT and the 2nd Finnish-Russian Mobile Linux Summit. Tampere, Finland, 24–27.
- 7. Chen, C.-F. (2006). Applying the analytical hierarchy process (AHP) approach to convention site selection. Journal of Travel Research, 45(2), 167–174.
- Chen, F., Wang, Q., Wei, Q., Ren, C., Shao, B., & Li, J. (2013). Integrate ERP system into business process management system. Proceedings of 2013 IEEE International Conference on Service Operations and Logistics, and Informatics, 436–439.
- 9. Daneva, M. (2004). ERP requirements engineering practice: lessons learned. IEEE Software, 21(2), 26–33.
- 10. Daneva, M., & Wieringa, R. (2010). Requirements engineering for enterprise systems: what we know and what we don't know? In Intentional perspectives on information systems engineering (pp. 115–136). Springer.
- 11. Dechsupa, C., Vatanawood, W., & Thongtak, A. (2018). Transformation of the BPMN Design Model into a Colored Petri Net Using the Partitioning Approach. IEEE Access, 6, 38421–38436. https://doi.org/10.1109/ACCESS.2018.2853669
- 12. der Aalst, W., & Stahl, C. (2011). Modeling business processes: a petri net-oriented approach. MIT press.
- 13. der Aalst, Wil M P. (1999). Formalization and verification of event-driven process chains. Information and Software Technology, 41(10), 639–650.
- 14. Dijkman, R. M., Dumas, M., & Ouyang, C. (n.d.). Formal Semantics and Analysis of BPMN Process Models using Petri Nets. 1–30.
- 15. Dunn, C. (2006). Business Process Modeling Approaches in the Context of Process Level Audit Risk Assessment: An Analysis and Comparison. International Journal of Accounting Information Systems, 2(7), 205–207.
- 16. Fortier, P., & Michel, H. (2003). Computer systems performance evaluation and prediction. Elsevier.
- Giaglis, G. (2014). A Taxonomy of Business Process Modeling and Information Systems Modeling Techniques A TAXONOMY OF BUSINESS PROCESS MODELLING AND. May. https://doi.org/10.1023/A
- 18. Haddara, M., & Elragal, A. (2013). ERP adoption cost factors identification and classification: a study in SMEs. International Journal of Information Systems and Project Management, 1(2), 5–21.
- 19. Halder, A., & Venkateswarlu, A. (2006). A study of petri nets modeling analysis and simulation. Department of Aerospace Engineering Indian Institute of Technology Kharagpur, India.
- Handfield, R., Walton, S. V, Sroufe, R., & Melnyk, S. A. (2002). Applying environmental criteria to supplier assessment: A study in the application of the Analytical Hierarchy Process. European Journal of Operational Research, 141(1), 70–87.
- 21. Heimdahl, M. P. E., & Leveson, N. G. (1996). Completeness and consistency in hierarchical statebased requirements. IEEE Transactions on Software Engineering, 22(6), 363–377.

- 22. Heitmeyer, C. L., Jeffords, R. D., & Labaw, B. G. (1996). Automated Consistency Checking of Requirements Specifications. 5(3), 231–261.
- 23. Helo, P., Anussornnitisarn, P., & Phusavat, K. (2008). Expectation and reality in ERP implementation: consultant and solution provider perspective. Industrial Management & Data Systems.
- 24. Kanchymalay, K., Krishnan, R., Arif, F., Amiruddin, S., Salam, S., & Hashim, U. R. (2013). The Extent of ERP Customization towards User Satisfaction in Daily Operation for Manufacturing Companies. JCP, 8(7), 1788–1792.
- 25. Klaus, H., Rosemann, M., & Gable, G. G. (2000). What is ERP? Information Systems Frontiers, 2(2), 141–162.
- 26. Murata, T. (1989). Petri nets: Properties, analysis and applications. Proceedings of the IEEE, 77(4), 541–580.
- Panorama Consulting Solutions, L. (2016). 2016 report on ERP systems and enterprise software: A Panorma consulting solutions research report. 2–30. https://www.panorama-consulting.com/wpcontent/uploads/2016/07/2016-ERP-Report-2.pdf
- 28. Partovi, F. Y., Burton, J., & Banerjee, A. (1990). Application of analytical hierarchy process in operations management. International Journal of Operations & Production Management.
- 29. Remco, M. (2008). Semantics and Analysis of Business Process Models in BPMN. 50(12). https://doi.org/10.1016/j.infsof.2008.02.006
- Röder, N., Wiesche, M., Schermann, M., & Krcmar, H. (2015). Workaround Aware Business Process Modeling. Wirtschaftsinformatik, 482–496.
- 31. Sackmann, A., Heiner, M., & Koch, I. (2006). Application of Petri net based analysis techniques to signal transduction pathways. BMC Bioinformatics, 7(1), 482.
- Search, H., Journals, C., Contact, A., Iopscience, M., Conf, I. O. P., & Address, I. P. (n.d.). A Comparative of business process modelling techniques. 012010. https://doi.org/10.1088/1757-899X/128/1/012010
- 33. Shaul, L., & Tauber, D. (2013). Critical success factors in enterprise resource planning systems: Review of the last decade. ACM Computing Surveys (CSUR), 45(4), 1–39.
- 34. Steps, F. (2013). Essential Process Modeling. https://doi.org/10.1007/978-3-642-33143-5
- 35. Tealab, A., & Galal-Edeen, G. (2007). Surveying Workflow Modeling Techniques. March.
- ur Rehman, T., Khan, M. N. A., & Riaz, N. (2013). Analysis of requirement engineering processes, tools/techniques and methodologies. International Journal of Information Technology and Computer Science (IJITCS), 5(3), 40.
- van Beijsterveld, J. A. A., & van Groenendaal, W. J. H. (2016). Solving misfits in ERP implementations by SMEs. Information Systems Journal, 26(4), 369–393. https://doi.org/10.1111/isj.12090
- 38. van der Aalst, Wil M P, & Stahl, C. (2011). Modeling business processes: a petri net-oriented approach. MIT press.
- 39. Wang, W., Ding, H., Dong, J., & Ren, C. (2006). Comparison of Business Process Modeling Methods. 1136-1141.
- 40. Zare, A., & Ravasan, A. Z. (2014). An extended framework for ERP post-implementation success assessment. Information Resources Management Journal, 27(4), 45–65. https://doi.org/10.4018/irmj.2014100103
- 41. Zowghi, D., & Gervasi, V. (2002). The Three Cs of requirements: consistency, completeness, and correctness. International Workshop on Requirements Engineering: Foundations for Software Quality, Essen, Germany: Essener Informatik Beitiage, 155–164.