Event-driven management algorithm of an Engineering documents circulation system

V Kuzenkov[1], A Zebzeev[2], E Gromakov[3]

[1] Graduate student of Tomsk Polytechnic University, CEO of JSC "TomskNIPIneft", Tomsk, 634050, Russia
[2] Graduate student of Tomsk Polytechnic University, lead engineer of JSC "TomskNIPIneft", Tomsk, 634050, Russia
[3] PhD, Tomsk Polytechnic University, Tomsk, 634050, Russia

Gromakov@tpu.ru

Abstract. Development methodology of an engineering documents circulation system in the design company is reviewed. Discrete event-driven automatic models using description algorithms of project management is offered. Petri net use for dynamic design of projects is offered.

1. Introduction

Project management in the oil-and-gas survey design companies (SDC) is an actual task. Various practices, in particularly PMI [1,2], are used for this. A complex flow of engineering documents is formed during the project activities execution process. The main disadvantages of the existing engineering documents circulation and design workflow management include the following:

- non-transparent of the project execution processes due to defect procedures of documentation process, distribution and keeping of documents, accounting documents;
- impossibility of modification process tracking of engineering document of design management in a multi-user environment SDC, complexity of detection and identification of insertion errors and difficulties to troubleshooting, respectively;
- impossibility of effective monitoring of the design process and objective assessment of the degree of the readiness of the project or its parts at a given time, including tracking their status within the life cycle and the location in the work process, as there is no centralized engineering project data management.

So, all SDC are constantly working on improving the engineering documents circulation system (EDCS): adaptation of packaged commercial products or development their own portal systems of design document management [3].

The aim of this work is evolution of EDCS development methodology in the design company independently. The following procedure for development of such EDCS is offered: generate detailed static (circuit) models describing the flow of project activities and information flows (engineering documents) accompanying their; develop a model of supervisory core which providing project activities management; convert these models into dynamic (coupled with the attributive settings of documents circulation in the company in ideal); use the results of the design for EDCS software coding. It can be expected that such a sequence of design will provide the necessary system project manageability.

2. Description of design process engineering documents circulation system in the design company

The suitable option of the such development beginning is eEPC-notation of ARIS CASE-package, which is well suited for building the static models used at all levels of project management. Such description is used in the quality management system of many subsidiaries oil-and-gas companies, in particular, of "TomskNIPIneft". Quite easy eEPC model can be converted to a dynamic model, in particular to the Petri net (PN), because it represents a strict model concept of interleaving of eventfulness and workflow.

Visual format of eERS-notation of projects design is a best practice, which providing understability and consistency of automaton model semantics of the SDC. In addition to the formal verification of discrete event systems (DES) control It provides semantic verification of the model description of the project activities through a collective agreement and refinement of the structural features and specifications context. As well as, such descriptions have an essential advantages over operational monitoring of requirement changes of specifications at execution of projects.

Model of project activities in the eEPC-notation can be presented by a finite set of events E of a finite set of functions F, a finite set of connectors C, a connector function T, which shows each connector to the appropriate type, and a finite set of directed arcs A [4]:

$$T \in C \rightarrow (or, xor, and), \tag{1}$$

$$A \subseteq (E \cup F) \cup (F \times E) \cup (E \times C) \cup (C \times E) \cup (F \times C)(C \cup F) \cup (C \times C). \tag{2}$$

The resulting EPC diagrams consist of three types of event units *E* of functions *F* and connectors *C*. Type of each connector is set by the function *T*. The ratio *A* is specified by the directed arcs, which connect event functions and connectors. This ratios are mutually disjoint (two functions and two events don't get connected to each other). Also, only events are the beginning and end of the diagram. Event description of the project activities allows to classify project management in the DES class [5]. Figure 1 shows eEPC model of works organization of preparation and issuance of tasks for project parts working out (TPPW).

As is seen from the model this notation describes not only operating sequence full but also document contents used in the EDCS operations. DES of this type can be described by the triple (*G, K, S*), where *G* – is the projects, *K* – requirements (specifications), establishing procedures for their implementation, and *S* – supervisor which must have the manageability property of the design process of the SDC under given specification constraints of projects performance. The functioning of such a DES is characterized by a set of events *E* in the project life cycle and management sequence e_c based on the events from *E*. A set of management sequences *S* form an attribute content languages of DES: the language *L(G)* generated by status of project *G*, the language *L(S)* generated by the supervisor *S*.

Description of the project activity in this notation allows for EDCS to set the event sequence path of project performance and by this means to formalize the DES specification. This allows to set a conceptual description of the engineering documents circulation in the SDC using language *L(G)* and *L(S)*. This formalization allows to determine sequence of control commands in the EDCS. This control commands are presented in accordance with the regulations of the DSC engineering documents. Engineering documents should be worked out by the project executors.

The EDCS structuring can start with the separation of all projects in the SDC. Let it be $G = (G^1, G^2, ..., G^n)$. For each G^i defined: a set of events that are supervisory commands E^i_c, a set of the expected events $E^i w$ that characterize the routine response of project activities (a set of expected reactions E^i_c to control of the project execution) and finally a set of events E^i_u is determined as external relative to G^i. Common alphabet for G^i is $E^i = (E^i_w, E^i_c, E^i_u)$.

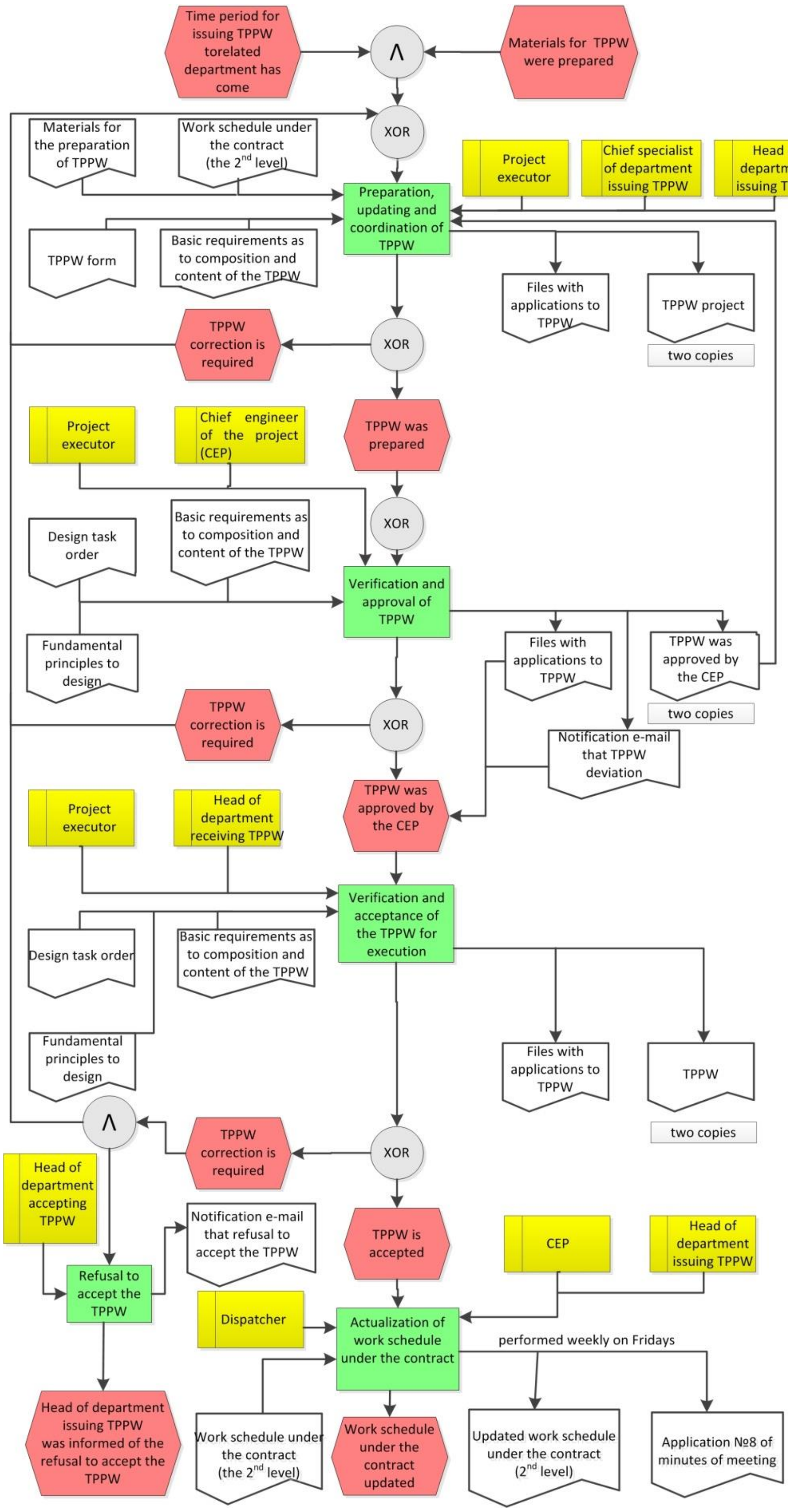

Figure 1. TPPW eEPC model.

Relation of the commands (E^i_c) of expected reactions and external conditions are usually presented in the form of diagrams in regulatory documents. It can be formalized by a finite state machine G^i [6]:

$$G^i = (Q^i, E^i, \delta, \Gamma^i, Q^i_m, q^i_0), \tag{3}$$

where:

Q^i – is a finite set of project states; E^i – is a set of project events; δ – is the transition function; Γ^i — is a set of acceptable events; $Q_m{}^i$ – is a set of states of Q (i.e. $\Gamma \subseteq Q$) called accept states; $q_0{}^i$ – is the start state, that is, the state of the automaton before any input has been processed, where $q_0 \in Q$.

As a result, at an early stage are determined: the composition of projects $G = (G^1, G^2 ... G^n)$ and a set of events E^i, each of which is structured on the sets $E^i{}_w$, $E^i{}_c$, $E^i{}_u$ of routine, managed and external events. Let's consider procedural supervisor module which should form the procedure of preparing and description TPPW, detection modification history of engineering document of design management in a multiuser environment of SDC, provide the necessary management by coordination and approval of projects.

This procedure can be formalized machine G^5 which shown in Figure 2.

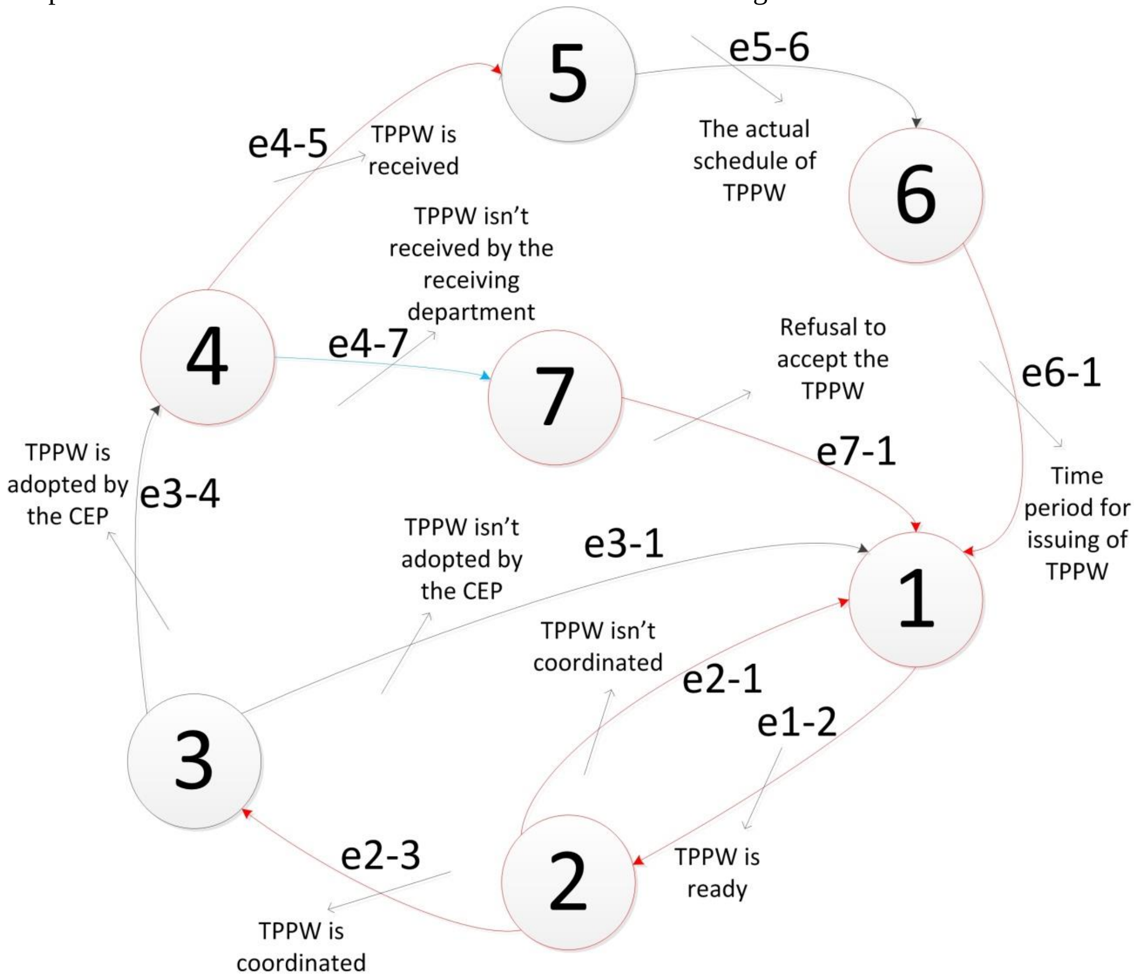

Figure 2. Transition graph of machine G^5.

The following states (from 1 to 7, respectively) generate in the machine: "TPPW preparation", "allows of TPPW in the department", "TPPW adoption by the chief engineer of the project (CEP)", "TPPW consideration by the receiving department", "updated work schedule under contract "," monitoring of TPPW dates", "TPPW deviation". As the graph shows, *e3-1, e3-4, e5-6* of a set of events E^i are related to the supervisory commands of the CEP. *e1-2, e2-1, e2-3, e4-1, e6-1, e7-1* are the expected events on the control commands. *e4-7* event does not related to the expected, because it usually appears after changing of external requirements to the project, unrecorded in preparation of TPPW. Automate description of project activities in the graph of projects eventness G^5 (Figure 2) allows to conclude that the engineering project documentation can be divided into several paths. One of them related to the conceptual type events (for example, changes in the project execution conditions, such as "*1-2-3-1*", "*1-2-3-4-7-1*"), the other - to the operating sequence of the project execution conditions ("*1-2-1*", "*1-2-3-4-5-6-1*"). From the *G*-automaton model of project activities of SDC follows that possible dead states of projects execution, related to stopping the project, can be solved by the changes of supervisor data and the status of the project or create a new project, which is ready to start.

According to our proposed methodology for the subsequent verification of the conceptual EDCS projects, its structural model must be converted to the PN notation, which will allow to establish manageability and observability of the decision. In order to reduce the PN dimensions and to minimize the number of possible system (structural) errors is logical to use operations patterns (standard samples) of the project activity [7,8]. During EDCS software development PN patterns are converted into object-oriented software modules that identify and show relation between classes or objects, without determining what the final classes or application objects to be used. The pattern denominates a solution the custom name that making it easier for communication developers between the individual EDCS modules, allowing to refer to the known patterns. Thus, due to the patterns, software component standardization is executed and the number of possible errors is reduced. Properly formulated design pattern allows to use it again and again after it finds a good solution. A set of operation patterns of the project activity in the SDC simplifies the EDCS developer a decision of the most suitable variant of its construction.

3. Summary

Automaton-oriented technology of the EDCS management in any (oil-and-gas, construction, energy or other) design company provides an understability and predictable during projects processes. The proposed algorithm preparation of the EDCS management will have consistensy property and will accord to established specifications (documented procedures in the SDC, operating procedures) as in supervisory management of projects portfolio as a management of portfolio individual projects.

References

[1] A Guide to the Project Management Body of Knowledge (PMBOK Guide) – Fifth Edition, Project Management Institute 2013.
[2] Ron Basu, Managing quality in projects: An empirical study, International Journal of Project Management 32 2014, pp. 178–187.
[3] Kuzenkov V, Zebzeev A, GromakovE., Resource leveling in the project design process by Petri net using, Advanced Materials Research Vol. 905 (2014) pp.752-756.
[4] Repin V.V., Eliferov V.G. Process approach to management. Modeling business processes. - M. Mann, Ivanov and Ferber 2013, 544 p.
[5] Christos G. Cassandras, Introduction to Discrete Event Systems. Second Edition. Springer Science & Business Media LLC 2008, 769 p.
[6] Linz, Peter. An Introduction to Formal Languages and Automata (5th ed.). Sudbury, MA: Jones & Bartlett Learning 2011, 437 p.
[7] N. Russell, A.H.M. ter Hofstede, W.M.P. van der Aalst, and N. Mulyar. Workflow Control-Flow Patterns: A Revised View. BPM Center Report BPM-06-22 2006, 134 p.
[8] W.M.P van der Aalst, A.H.M. ter Hofstede, B. Kiepuszewski, and A.P. Barros. Workflow Patterns. Distributed and Parallel Databases, **14(3),** p. 5-51, July 2003