Available online at www.sciencedirect.com

ScienceDirect

IFAC Papers Online CONFERENCE PAPER ARCHIVE

IFAC PapersOnLine 50-1 (2017) 10526–10531

Functional Programming for Business Process Modeling

Abhishek SAINI * **Laurent THIRY** *

* *Ecole Nationale Supérieure d'Ingènieurs Sud Alsace 12, rue des freres Lumière, 68200 Mulhouse (e-mail: abhishek.saini@uha.fr, laurent.thiry@uha.fr)*

Abstract: This paper presents how Functional Programming (FP) helps to provide an other formal semantics (relation between the syntax and the model of computation) for Business Process Modeling (BPM); a semantics relatively different from Object Oriented semantics. More precisely, it proposes a general methodology to model business processes using mathematical functions and higher-order functions. We describe the basic part of Business Process Modeling, behavioral semantics via Petri Nets (PN) and Functional implementation of the models. Also, we will see how the business process model is translated into its equivalent form in Petri Nets and how these can be described through Functional Programming.

Keywords: Functional Programming, Semantics, Business Process Modeling, Petri Nets.

1. INTRODUCTION

The term "Business Process Modeling" was coined in the 1960s in the area of systems engineering by S. Williams and until the 1990s, it became popular. BPM is the way of describing processes within enterprises, so that a model (abstract representation which can be manipulated) can be analyzed and improved, (Aguilar-Saven, 2004). A business process is viewed as a set of related, structured activities or tasks within an organization whose objective is to produce a specific product or service. A task needs to be finished before its deadline or in a definite time to work towards goal. There are many techniques to model processes such as a flow chart, functional flow block, Unified Modeling Language (UML), Business Process Model and Notations (BPMN), etc.

Functional Programming (FP) is a style of programming in which programs are executed by evaluating expressions. Functional Programming focuses on simplicity and genericity, (Hughes, 1989), functions are considered first-class, which states that they can be passed as parameters to other functions or be returned as a result of a function. FP does computation as the evaluation of mathematical functions without changing state and mutating data. In comparison to Object-Oriented Programming (OOP), order of execution is less important, functions with no side-effects, uses recursion concept to iterate collection data, supports both "Abstraction over Data" and "Abstraction over Behavior", etc.

Semantics is the study of meaning, (Lyons, 1977). In terms of programming language theory, semantics is the field related with the mathematical study of the meaning of programming languages. It describes the processes a computer follows when executing a program in that specific language. This can be viewed by describing the relationship between the input and output of a program, or an execution of how the program will execute on a certain platform, developing a model of computation. In this paper, Functional Programming has been used to propose a formal semantics of business model.

Many research papers have been presented and discuss about the concepts of Business Process Modeling with Petri Nets or Workflow nets or Object-oriented but a few are based on Functional Programming. In particular, iTask system (iTasks) is a task-oriented programming toolkit for programming workflow support application in Clean.

The paper is organized into three sections as follows: The next section discusses and introduces Business Process Modeling and Functional Programming. Our proposition is presented in Section 3. Finally, Section 4 concludes this paper by summarizing the main points introduced in the paper and its future aspects.

2. CONCEPTS AND STATE OF THE ART

2.1 Business Process Modeling

A business process is a set of tasks and resources required to achieve some services. It is also stated as a set of activities that once completed will accomplish goal. There are constraints and rules that have to be met. Basically, there are three types of business process:

(1) Management processes that govern the operations of a production system.
(2) Operational processes that constitute the core business and create the primary value stream.
(3) Supporting processes which supports the core processes.

A business process can be decomposed into several sub processes, with specific attributes. A process model is a representation (graphical or textual) of business processes represented as a set of sequential or parallel process activities combined together to achieve a common goal. Using

10.1016/j.ifacol.2017.08.1299

a model, it becomes possible to find out how the system will behave and the properties it will acquire. There are different modeling languages that are used to model business processes such as BPMN, (Herden et al., 2015), suitability of the BPMN for Business Process Modeling using workflow patterns, (Wohed et al., 2006). Modeling business process through activity diagrams patterns, (Andre et al., 2014). (Mili et al., 2010) describes the classification of business process modeling languages. The two most used graphical notation for business processes are BPMN and Unified Modeling Language Activity Diagram (UML AD) which are discussed in (Geambasu, 2012).

As an example, the Figure 1 models the behavior of the ATM machine for withdrawing money with the Business Process Modeling Language (BMPL).

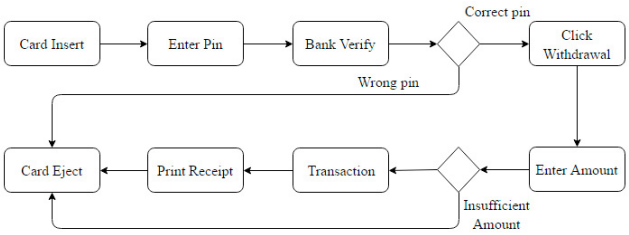

Fig. 1. Business Process Modeling ATM.

The above model shows how the data flow between the different activities (represented as boxes), how conditions take places and how the flow of control takes place. The activities have some data associated with them, the data related to the current activity and its result will act as an input to the next activity. The arrow shows the transition between the two activities which involves the transfer of data and also the flow of control between activities and it also describes the change in the state of the activities. The diamond is used as a condition which has two outputs and the selection of the output is depend on the output of the previous activity which acts as an input to the condition.

The different semantics that are used to give a behavioral description of the business process model are Petri Nets or Workflow nets. A Petri Net is represented as a bipartite graph that have tuple $N = (P, T, A)$ where P is the set of places, T is the set of transitions and $A \subset (P \times T) \cup (T \times P)$ is the set of flow relations, (Zhang et al., 2010). They are the basic model of parallel and distributed system. They were documented by Carl Adam Petri in 1962 in his PhD thesis "Kommunikation mit Automaten". The basic idea behind it, is to draw the changes of state with transitions in a system.

Petri Net is a strong technique for modeling and analysis of the system behavior where resource sharing, concurrency and synchronization are a significant matter to take into account, (Reisig, 1985). A classical Petri Net is that which consists of place nodes holding resources, the number of resources is denoted by the number of anonymous tokens on the places, transition nodes consuming and producing resources, and arcs between the nodes specifying dependencies between transitions and places with resources on them. A transition is only enabled if all the input places have sufficient tokens, and an enabled transition can go off by consuming tokens from all the input arcs while synchronously producing tokens along each output arcs.

The Figure 2 gives an example of Petri Net which contains places ($S1...S7$), place contains tokens and transitions ($T1...T7$) that are joined by directed arcs.

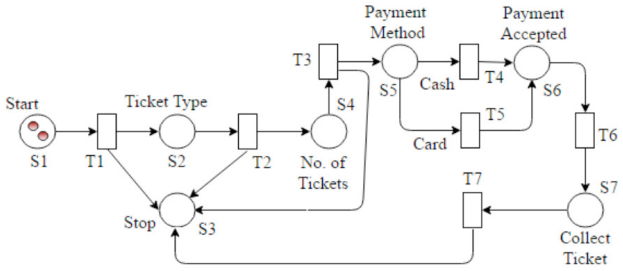

Fig. 2. Petri Net of Getting Tram Ticket.

For a basic purpose, it is sufficient to consider classical Petri Nets to model business process, (van Hee et al., 2013). But sometimes to evaluate process, the classical Petri Nets are not sufficient, that's why the extension of Petri Nets are needed that are know as high-level Petri Nets, especially Coloured Petri Nets (CPN), (Jensen and Kristensen, 2009). In CPN, places are defined with the type that is colour set of tokens, tokens and resources are replaced by individual (or coloured) tokens defined with typed values of some inscription language, sometimes at same place multiple tokens may be found. Transitions and arcs are defined by program code that operates on the token values. Incoming arcs are annotated with variables or patterns which has a code to the transition and to the output arcs. Transition is also parameterized by the values of input tokens, can have additional binding values, can have guards (or conditions) that determine whether the given set of tokens is sufficient for firing of the transition, (Reinke, 2000).

A workflow net is a Petri Net that has two special nodes, the first node is known as starting node that is start place and the other node is known as final node that is end place. In a Worflow net every transition is on a path from start to end place. (van der Aalst, 1998) introduced worflow management as an application domain for Petri Nets. The Figure 3 gives an example of Workflow net.

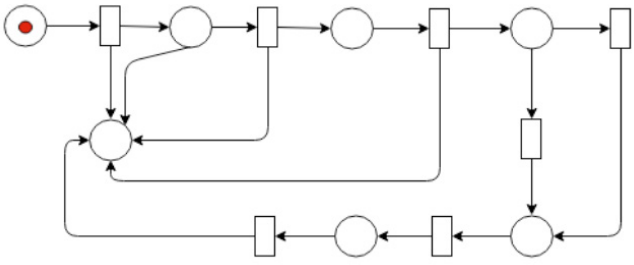

Fig. 3. Workflow net.

A workflow is a representation of a sequence of activities, declared as a work of persons, of a simple or complex mechanism, or machines. A workflow can be considered a view or picturization of real work. The flow may refer refer to a document, services or a product that is being