

Formal Modeling and Verification of Virtual Community Systems

Elthon Oliveira

Federal University of Alagoas, Campus Arapiraca, Brazil

Hyggo Almeida

Federal University of Campina Grande, Brazil

Leandro Silva

Federal University of Campina Grande, Brazil

Nadia Milena

Federal University of Campina Grande, Brazil

Frederico Bublitz

Federal University of Campina Grande, Brazil

Angelo Perkusich

Federal University of Campina Grande, Brazil

INTRODUCTION

In the last years, virtual community systems (VCS) (Bublitz, Barbosa, & Costa, 2004) have been used as one of the main mechanisms for communication and collaboration among people throughout the world--the people whom use this kind of system compose the so-called virtual community. Several systems providing different features and tools such as forums, e-mails, and videoconference, among others, represent a revolution in the way that people interact with others.

Depending on the kind of virtual organization and on the role played by a VCS in such an organization, system reliability becomes essential. For example, enterprise organizations can use a VCS as the main environment for exchanging experiences, discussing projects, and evaluating proposals. Such information managed by the virtual community system is very important and essential for the enterprise sustainability and progress.

Due to the importance of the information stored by these systems, an error can cause time and money to be lost. Errors can be introduced during the implementation phase or the system architecture design. It is known that as soon as errors are detected, fewer efforts are needed to correct them. Considering software engineering techniques, although developing software systems based on reusable components has been pointed

as suitable for achieving flexibility, maintenance, and evolution (McInnis, 1999), it does not assure security and trustworthiness.

To solve this problem, formal modeling and verification techniques and tools have been used in large scale in the context of software engineering. As it was said in Parnas (1996), "Professional engineers are expected to use discipline, science, and mathematics to assure that their products are reliable and robust." Different from testing techniques, formal verification guarantees that the specification of a system is correct for all possible behavior, making the system more reliable.

In this article, we describe the application of a component-based framework named COMPOR-CPN (Almeida et al., 2005) to the formal modeling and verification of virtual community systems. This framework is a Petri Net (Murata, 1989) modeling of the COMPOR component model specification (CMS) (Almeida et al., 2006), and makes easier the activities related to the modeling and verification of VMCs by using a component-based approach. In fact, to be more precisely, it is a Coloured Petri Net (CPN) (Jensen, 1992, 1997) modeling of CMS.

The remainder of this article is organized as follows. In the background section, a brief and informal introduction to CPN is presented. In the COMPOR-CPN section, the framework COMPOR-CPN is introduced. The modeling and verification of the virtual

community system is shown in the virtual community system modeling and verification section. Finally, in the conclusion and future trends section, some future trends are discussed as well as final remarks.

BACKGROUND

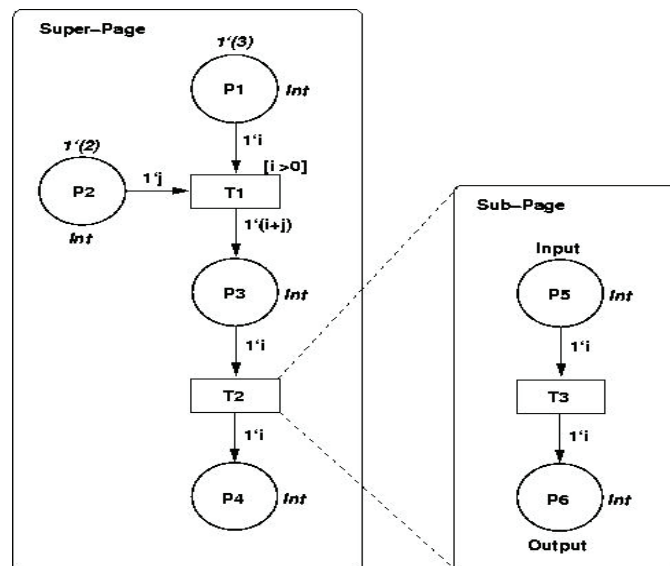
Coloured Petri Nets (CPN) (Jensen 1992, 1997) are a formal method with a mathematical base and a graphical notation for the specification and analysis of systems with characteristics such as concurrent, parallel, distributed, asynchronous, timed, among others. For the mathematical definition, the reader can refer to Jensen (1992, 1997). The graphical notation is a bipartite graph with places represented as ellipses, and transitions represented as rectangles. Transitions represent actions and the marking of the places represent the state of the model. A marking of a place at a given moment are the tokens present at that place. A token can be a complex data type in CPN/ML language (Christensen & Haagh, 1996). Each place has an associated color set that represents the kind of tokens the place can have. Transitions can have guards and code associated to it. Guards are boolean expression that must be true for the transition to fire. Code can be a function that is executed every time the transition fires. Arcs go from places to transitions and from transitions to places and never from transition to transition or place to place,

and can have complicated expressions and function calls associated to it.

For a transition to fire, it is necessary that all input places, that is, places that have arcs that go from the place to the transition, have the number of tokens greater than or equal to the weight of the arc, $w(p,t)$, and the guard of the transition must be true. When these characteristics hold the transition, is said to be enabled to fire. An enabled transition can fire at any time and not necessarily immediately. Once a transition fires, it removes $w(p_i,t)$ from each input place p_i , and the output places, that is, places that have arcs from the transition to the place, receive tokens according to the arc expression from the transition to the place: $w(t,p)$. This expression is evaluated to an integer number that is the weight of the arc.

A CPN model can also have a hierarchy (CPN Hierarchy, 2007). Two mechanisms, *substitution transitions* and *fusion places*, can be used to structure the model in a more organized way. Substitution transitions are transitions that represent another CPN model called page. The page where the transition is placed is called super-page and the page represented by the transition is called sub-page (Figure 1). In the sub-page, there are places that can be input, output, or input/output places. These places are associated to input and output places of the transition in the super-page called sockets. Fusion places are places that are physically different but always have the same marking. A change in the marking

Figure 1. HCPN model



6 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: www.igi-global.com/chapter/formal-modeling-verification-virtual-community/17667

Related Content

Gendered Experiences of Mobile Gaming and Augmented Reality: Engagement with Pokémon Go among University Students

William Goette, Julie A. Delello and Rochell R. McWhorter (2019). *International Journal of Virtual and Augmented Reality* (pp. 54-67).

www.irma-international.org/article/gendered-experiences-of-mobile-gaming-and-augmented-reality/239898

Technology: A Tool for Creating Collaborative Learning Environments

Rita Gravina and Helena Pereira-Raso (2013). *Cases on Online Learning Communities and Beyond: Investigations and Applications* (pp. 135-149).

www.irma-international.org/chapter/technology-tool-creating-collaborative-learning/68118

Improving Virtual Teams through Creativity

Teresa Torres-Coronas and Mila Gasco-Hernandez (2008). *Virtual Technologies: Concepts, Methodologies, Tools, and Applications* (pp. 1679-1686).

www.irma-international.org/chapter/improving-virtual-teams-through-creativity/31014

A Preliminary Investigation Into the Effects of Gamified Virtual Reality on Exercise Adherence, Perceived Exertion, and Health

Katherine Jane Hoolahan (2020). *International Journal of Virtual and Augmented Reality* (pp. 14-31).

www.irma-international.org/article/a-preliminary-investigation-into-the-effects-of-gamified-virtual-reality-on-exercise-adherence-perceived-exertion-and-health/283063

Cubios Transreality Puzzle as a Mixed Reality Object

Ilya V. Osipov (2017). *International Journal of Virtual and Augmented Reality* (pp. 1-17).

www.irma-international.org/article/cubios-transreality-puzzle-as-a-mixed-reality-object/188478