Model reservation resource of computer complexes system

Dmitry Kornev[1], Evgeny Nikulchev[2*]

[1]*Scientific Research Institute for Automation and Communication, Nizhny Tagansky str., 27, p. 1 Moscow, Russia*
[2]*Moscow Technology Institute, Leninsky Prospect, 38a, Moscow, Russia*

[*]*Corresponding author's e-mail: nikulchev@mail.ru*

Received 19 February 2017, www.cmnt.lv

Abstract

The article presents a method for calculating redundancy systems for platform management and organization of the movement of high-speed trains. The model is developed on the mathematical formalism of Petri nets. Calculated estimated time system failover. Proved the feasibility of using a majority redundancy.

Keywords:

information system,
Petri net,
information complex,
reservation of computer complex

1 Introduction

All high-tech systems require means of maintaining operability during sudden failures. This is particularly important for transport systems. Traffic safety is the main principle of the transport systems and provides high efficiency of the railways.

Currently, Russia's specialists creating an information platform, "High-speed rail". The platform is designed to traffic management of high-speed trains. The reliability of the platform should be provided with additional computing resources. These resources will be included with the sudden failure of platform elements.

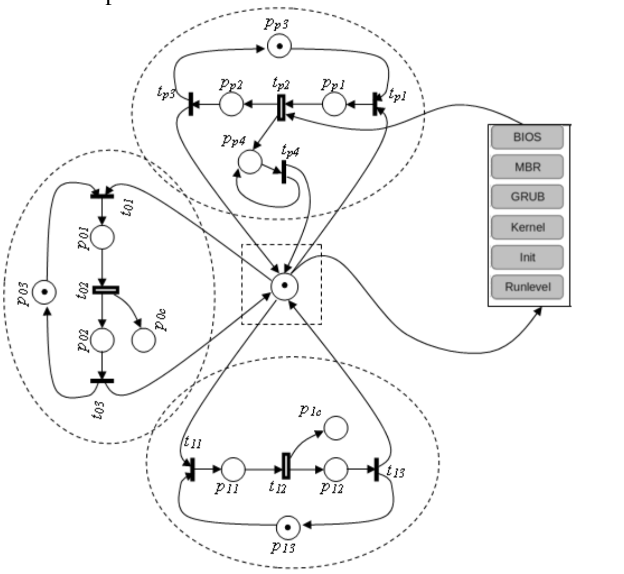

FIGURE 1 Model of interaction of the Platform and the VM in the event of failure of computing resources

Information resource characterized by the time ready to work. This time should match the characteristics of the control object. Platform deployed on virtual machines (VMs). These VMs enable communication platforms and each locomotive. Objects Management Platform is the system management and control locomotive. Information exchange interval "platform locomotive" is $\Delta T = 100$ ms.

The timing of the inclusion of additional resources Platforms τp r carried out on the model of interaction between the host and VMs. Host constantly interacts with multiple VMs. The model used by the mathematical formalism of colored Petri nets (Figure 1).

2 Mathematical model

Object's model to the mathematical formalism of Petri net has the form [1, 2]:

$$\Pi = \{P, T, I, O, \mu\}, \quad (1)$$

$P = \{p_1, p_2, \dots, p_i, \dots, p_n\}$ - a plurality of positions;
$T = \{t_1, t_2, \dots, t_j, \dots, t_m\}$ - a plurality of transitions;
I - input function of transitions (this value determines the multiplicity of input arcs of transitions (t_j));
O - output function of transitions (this value determines the multiplicity of output arcs of transitions $O(t_j)$);
μ - marker.

Model of system "Platform-locomotive" is presented as a graph with two types of vertex - positions and transitions. Positions and transitions are connected by arcs.

Routes in the model represented by the equations:

$$|I(p_i)| = \left|\{t_j | p_i \in O(t_j)\}\right| = 1, \quad (2)$$

$$|O(p_i)| = \left|\{t_j | p_i \in I(t_j)\}\right| = 1, \quad (3)$$

$\{t_j | p_i \in O(t_j)\}$ - set of transitions, where p_i is the output
$\{t_j | p_i \in I(t_j)\}$ - set of transitions, where p_i is the input.
Transition $t_i \in T$ performed under the condition [3]

$$t_j: \mu(p_i) \geq \#(p_i, I(t_j)) \quad \forall p_i \in P, \quad (4)$$

$(p_i, I(t_j))$ - the multiplicity of the input position p_i of transition t_j.

After the transition $t_i \in T$ position p_i receives a new marker μ':

$$\mu'(p_i) = \mu(p_i) - \#(p_i, I(t_j)) + \#(p_i, O(t_j)). \quad (5)$$

Figure 1 shows:

- p_0 – resource Host Platforms;
- $p_{01}, \ldots, p_{k3}$ - the current state of the information environment k VM:
- $p_{01}, \ldots, p_{k1}$- resource of the Host; $p_{02}, \ldots, p_{k2}$ - it frees the host system resources; $p_{03}, \ldots, p_{k3}$- waiting host resource ; $p_{0c}, \ldots, p_{kc}$- counter operations;

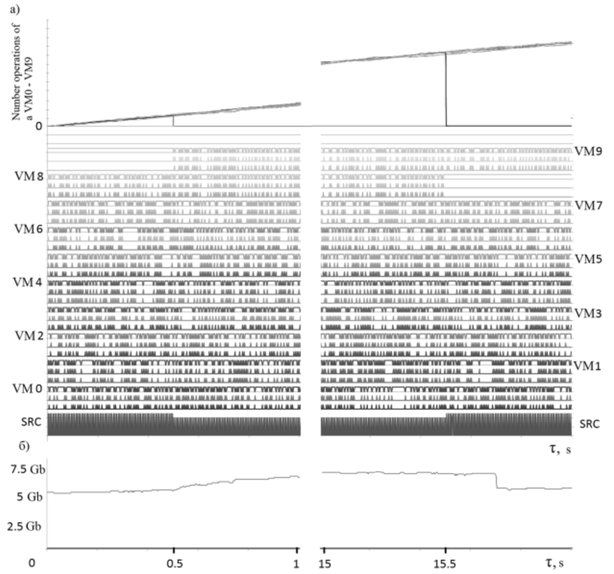

FIGURE 2 Deploying backup VM9: Simulation results (a); Experiment (b)

The mathematical model contains an additional structure [4, 5]:

$$P_{\mathrm{p}} = \{p_{\mathrm{p}1}, p_{\mathrm{p}2}, p_{\mathrm{p}3}, p_{\mathrm{p}4}\}, \quad (6)$$

$t_{01}, \ldots, t_{k3}$ - distribution of host resources and the generation of k VM requests: $t_{01}, \ldots, t_{k1}$ - providing a Host of resources ; $t_{02}, \ldots, t_{k2}$- working with Host resources; $t_{03}, \ldots, t_{k3}$ - host resources return.

$$T_{\mathrm{p}} = \{t_{\mathrm{p}1}, t_{\mathrm{p}2}, t_{\mathrm{p}3}, t_{\mathrm{p}4}\}, \quad (7)$$

p_{p1}- provide resource to reserve; p_{p2} - the release of the resource of reserve; p_{p3} - waiting for the resource for reserve; t_{p1} - providing resource to reserve; t_{p2} - work with the resource reserve host; t_{p3} - return resource of reserve.

3 Simulation

This structure is an informational reserve host. $p_{\mathrm{p}4}$ position provides job reserve. The memory of reserve sequentially loaded operating system modules: BIOS, MBR, GRUB, core, Init, Runlevel; this process reflects the transition t_{p2}. In the calculations of module loading time correspond to OS load time computer-based Core i3 Duo. Transition t_{p4} closes fully loaded software in the host memory reserve. At this moment the reserve receives information from the VM.

The calculation was carried out provided that: requests from locomotives ΔT, and one request service time τ_{max} had a uniform distribution with restrictions ΔT = 100 ms and ms τ_{max} = 99,5.

Figure 2a illustrates the functioning of the Platform in terms of Petri nets. The platform serves computers nine locomotives (VM0-VM8).

When τ = 0,5 s. is error on VM8 session on the Platform. At this point it begins to boot VM9 reserve. This is a time τ_p = 15s. Where τ = 15,5 to reserve loading over. Control is passed from VM8 to VM9. Similar experiments were conducted on actual host. Time unfolding real reserve was τ_p = 11 - 17. for different download host (Figure 2b, Table 1).

TABLE 1 Time deployment of VM backup

Software	time τ_p, s
Loading the virtual machine to the base operating system	7-12 s.
Loading virtual machine with snapshots	11-17 s.
Loading database	8-12 s.
Loading software with automatic driving	4-6 s.

4 Results and conclusions

The simulation results showed good agreement between calculated and experimental values τ_p time.

Modeling was carried out and on the Platform to deploy the reserve only with the basic software. In this case, the real time deployment of the reserve (VM9) is τ_p = 7 - 12. The calculated and experimental results unfolding VM9 with basic software are shown in Figure 3 and Table 1.

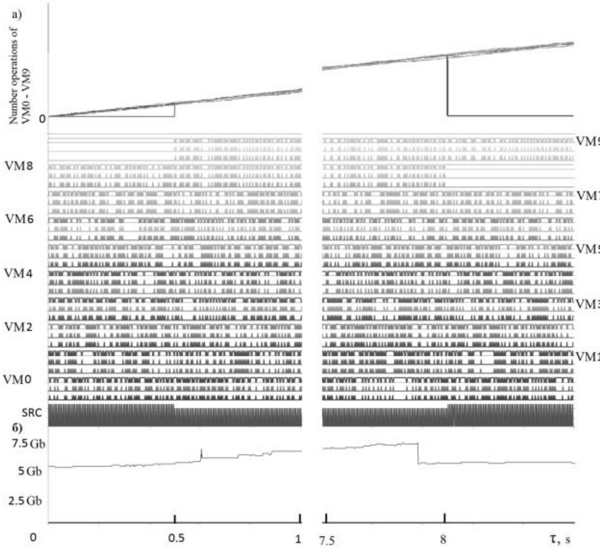

FIGURE 3 Deploying backup VM9 with basic software: simulation results (a); Experiment (b)

Modeling it possible to calculate the minimum reserve time of deployment on the platform. It is time $\tau_p \geq 5$s. For Platform train control system $\tau_p \geq 5$c is too large. The platform should deploy more VMs on the host, and give her control function of T2 = 100 ms.

This justifies the use for the Platform for the majority system redundancy.

Reference

[1] Kotov V 1984 *Petri nets* M.: Science: Home edition of Physical and Mathematical Literature 160 p.

[2] Leskin A 1989 *Petri nets in modeling and management* / A A Lesquin, P A Maltsev A M Spiridonov L.: Science 133 p.

[3] Zaitsev D 2013 Toward the Minimal Universal Petri Net *IEEE Transactions on Systems, Man, and Cybernetics: Systems* **144** 1–12

[4] Kornev D 2014 Study of algorithms of information systems using the mathematical formalism of Petri nets *Electronic Journal Cloud of*

Science **1**(1)

[5] Hee K van., Sidorova N M 2003 *Soundness and separability of workflow nets in the stepwise refinement approach* In van der Aalst, W. M. P.; Best, E. Voorhoeve Application and Theory of Petri Nets. Lect Notes in Comput Sci. 2678. Springer. pp. 337–56

Authors

	**Dmitry Kornev 08.03.1989, Russia** **Current position, grades**: Candidate of Engineering Sciences. He graduated from the Moscow State University of Railway Engineering (MIIT) by specialty "Computers, Systems and Networks" in 2011. He got a degree in technical sciences in 2015. **Publications**: 4 papers in peer-reviewed journals. Scientific interests: Information technologies. **Experience**: Worked as engineer in Scientific Research Institute for Automation and Communication
	**Evgeny Nikulchev** **Current position, grades**: vice-rector and professor of Moscow Technological Institute **University studies**: Dr of Sci. (Nonlinear Dynamics, 2006), PhD (Computer Science, 2000) **Scientific interest**: big information systems design and management, management theory, scientific research management **Publications**: More than 40 publications in the computing services, cloud computing, artificial intelligence. **Experience**: It is artwork in the development of computing services for software applications for different purposes. He was Chairman and member of the scientific committees of international conferences on information systems, e-learning; chief Editor of the journal Cloud of Science, guest editor of several journals.